

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Programa de Pós-Graduação em Engenharia Elétrica

Protocolo de Roteamento Multi-Métrico para Redes de Sensores Sem Fio – CA-AODV

Carlos Wagner Barros Araújo

Campina Grande
12 de Novembro de 2009

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Programa de Pós-Graduação em Engenharia Elétrica

Carlos Wagner Barros Araújo

Protocolo de Roteamento Multi-Métrico para Redes de Sensores Sem Fio – CA-AODV

Dissertação submetida à Coordenação do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande - Campus de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Mestre em ciências no Domínio de Engenharia Elétrica (MSc).

Área de concentração: Instrumentação eletrônica

José Sérgio da Rocha Neto

Orientador

Angelo Perkusich

Orientador

Campina Grande , Paraíba, Brasil

©Carlos Wagner Barros Araújo, 12 de Novembro de 2009

UFCG - BIBLIOTECA

UFCG - BIBLIOTECA - CAMPUS I	
812	24-08-020

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL DA UFCG

A662p

2009

Araújo, Carlos Wagner Barros.

Protocolo de roteamento multi-métrico para redes de sensores sem fio
– CA-AODV / Carlos Wagner Barros Araújo. – Campina Grande, 2009.
75 p. il.

Dissertação (Mestrado em Engenharia Elétrica) – Universidade
Federal de Campina Grande, Centro de Engenharia Elétrica e Informática.
Referências.

Orientadores: Prof. Dr. José Sérgio da Rocha Neto, Prof. Dr.
Angelo Perkusich.

1. Telemidição. 2. Redes ad hoc sem fio. 3. Roteamento. 4. Padrão
IEEE 802.15.4. I. Título.

CDU – 621.398(043)

**PROTOCOLO DE ROTEAMENTO MULTI-MÉTRICO PARA REDES DE
SENSORES SEM FIO (CA-AODV)**

CARLOS WAGNER BARROS ARAÚJO

Dissertação Aprovada em 23.10.2009



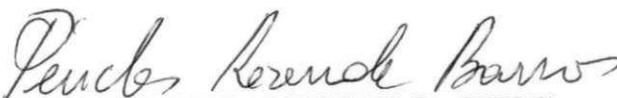
JOSÉ SÉRGIO DA ROCHA NETO, D.Sc., UFCG
Orientador



ANGELO PERKUSICH, D.Sc., UFCG
Orientador



HYGGO OLIVEIRA DE ALMEIDA, D.Sc., UFCG
Componente da Banca



PÉRICLES REZENDE BARROS, Ph.D., UFCG
Componente da Banca

CAMPINA GRANDE - PB
OUTUBRO - 2009

*Aos meus pais, Afonso Araújo Neto e Maria de Fátima
Barros Araújo, pelas angústias e preocupações que
passavam por minha causa, por terem dedicado suas vidas
a mim, pelo amor, carinho e estímulo que me ofereceram
e pelo exemplo de vida, dedico-lhes esta conquista como
gradidão, que sempre estiveram presentes nos momentos
mais difíceis, colaboradores e incentivadores, minha
grande fonte de alegria.*

Resumo

A fim de atender à crescente tendência de acesso ubíquo à informação novas arquiteturas flexíveis de redes estão emergindo. Essas arquiteturas são referidas como redes *ad hoc* sem fio. Uma rede *ad hoc* sem fio é uma coleção de nós aglomerados criando dinamicamente uma rede, sem estrutura fixa nem gerenciamento centralizado. Tais redes são caracterizadas por uma topologia dinâmica e por possuírem limitações de energia, processamento e memória nos seus dispositivos. Dada as características dinâmicas das condições do meio, principalmente em ambientes *indoor*, essas redes possuem grande dificuldade em dar garantias aos requisitos de desempenho. Recentemente, com intuito de incrementar a eficiência na entrega dos dados, têm surgido diversos trabalhos relacionados ao desenvolvimento de protocolos de roteamento que tomam suas decisões baseadas em métricas de qualidade das rotas. A maioria desses trabalhos estão relacionadas ao uso de um modelo *cross-layer*, no qual as diversas camadas de protocolo interagem entre si. O interesse deste projeto é o estudo de uma métrica de qualidade de rotas e seu emprego no desenvolvimento de um protocolo de roteamento para redes de sensores sem fio (RSSF), utilizando o padrão IEEE 802.15.4.

Palavra-chave: Redes *ad hoc* sem fio, Roteamento, Padrão IEEE 802.15.4.

Abstract

Nowadays, the increasing trend of ubiquitous access to information has created necessity for new flexible network architectures. These new architectures are referred to as wireless ad hoc networks. A wireless ad hoc networks is a collection of nodes, wich creates dynamically a network with no fixed structure nor centralized management. Such networks are characterized by a dynamic topology and by devices with limited resources (energy, processing capacity and memory). Due to the dynamic conditions of the environment, especially indoor environments, it's very difficult to assure performance in these networks. Recently, in order to increase efficiency in data delivery, there have been several works related to the development of routing protocols that make their decisions based on routes quality metrics. The most part of these works are related to the use of a cross-layer architecture. The cross-layer architectures employs interactions between the protocol layers to obtain quality informations. The interest of this project is the study of a quality metric for the routes and use it on the development of a routing protocol for wireless sensor networks (WSN), using the IEEE 802.15.4 standard.

Keywords: Wireless ad hoc networks, Routing, IEEE 802.15.4-2003 standard.

Agradecimentos

- A Deus que é Esperança, Paz, Harmonia, a Verdadeira Força, que nos faz e nos dá tudo que temos e somos;
- Aos meus pais Afonso Araújo Neto e Maria de Fátima Barros Araújo, grandes timoneiros que sempre me guiaram pelos ditames do amor, carinho, otimismo, perseverança, honestidade, a quem sem dúvida alguma, devo tudo que tenho e sou;
- Aos meus grandes amigos conquistados no laboratório de elétrica e tantos outros, pelo o apoio, incentivo e sobretudo, por estarem sempre presentes diante da menor solicitação de suas presenças;
- Aos meus orientadores José Sérgio da Rocha Neto e Angelo Perkusich pela paciência que sempre souberam me encaminhar nos estudos;
- À CAPES, por financiar este trabalho.

Conteúdo

Lista de Figuras	p. viii
Lista de Tabelas	p. x
Lista de Abreviaturas e Símbolos	p. 1
1 Introdução	p. 1
1.1 Problemática	p. 2
1.2 Objetivos do Trabalho	p. 3
1.3 Relevância	p. 4
1.4 Estrutura da Dissertação	p. 4
2 Redes de Sensores Sem Fio (RSSF)	p. 6
2.1 Controle de Acesso ao Meio em Redes Sem Fio	p. 8
2.2 Roteamento em RSSF	p. 9
2.3 Otimização da Pilha de Protocolos	p. 11
2.4 Conclusão	p. 13
3 Estudo da Métrica de Qualidade Aplicada no Roteamento em RSSF	p. 14
3.1 Métrica de Qualidade	p. 14
3.2 Protocolo Proposto	p. 17
3.3 Método Estatístico para Análise dos Resultados	p. 19
3.4 Simulações	p. 20
3.5 Conclusão	p. 22

4	Ad-Hoc On-demand Distance Vector (AODV)	p. 24
4.1	Descoberta de Rotas	p. 25
4.2	Manutenção de Rota	p. 27
4.3	Conclusão	p. 28
5	<i>Congestion-Aware AODV (CA-AODV)</i>	p. 29
5.1	Descoberta de Rotas	p. 29
5.2	Mecanismos: Prevenção Contra <i>Loops</i> e Detecção e Eliminação de <i>Loops</i>	p. 35
5.3	Mecanismo de Análise das Rotas	p. 37
5.3.1	Parâmetros de Disponibilidade	p. 38
5.3.2	Composição da Métrica de Disponibilidade	p. 38
5.3.3	Composição da Métrica Usando Modelo FUZZY	p. 40
5.4	Conclusão	p. 43
6	Simulações: Resultados e Discussões	p. 45
6.1	Resultados das Simulações	p. 46
6.1.1	Cenário 1	p. 46
6.1.2	Cenário 2	p. 48
6.1.3	Cenário 3	p. 50
6.1.4	Cenário 4	p. 52
6.2	Conclusão	p. 53
7	Implementação e Testes do CA-AODV numa Plataforma Real	p. 55
7.1	Módulo XBee	p. 55
7.2	ADuC8XX	p. 57
7.3	Implementação do CA-AODV	p. 58
7.3.1	Nós Fonte	p. 59
7.3.2	Nós Destino	p. 60

7.3.3	Nós Roteadores	p. 62
7.4	Testes da Implementação	p. 67
7.5	Conclusão	p. 68
8	Conclusões	p. 69
8.1	Sugestões para Trabalhos Futuros	p. 71
	Referências	p. 72

Lista de Figuras

1	Representação de uma RSSF	p. 3
2	Pilha IEEE 802.15.4	p. 10
3	Diagrama de uma rede <i>ad hoc</i> sem fio com uma rota	p. 15
4	Diagrama de uma <i>ad hoc</i> sem fio com duas rotas	p. 15
5	Cenário de avaliação da métrica	p. 21
6	Comparativo curvas do atraso do pacote ACK para Exp1 (sem congestionamento) e Exp2 (com congestionamento)	p. 22
7	Comparativo curvas do atraso do pacote ACK para Exp1 (sem congestionamento) e Exp3 (sem congestionamento)	p. 23
8	Descoberta de rota	p. 26
9	Representação do pacote RREQ do AODV	p. 26
10	Representação do pacote RREP do AODV	p. 27
11	Descoberta de rotas	p. 30
12	Representação do pacote RREQ do CA-AODV	p. 30
13	Rede com duas rotas	p. 33
14	Representação do pacote RREP do CA-AODV	p. 34
15	Tabela de roteamento do CA-AODV	p. 34
16	Formação de <i>loop</i>	p. 35
17	Primeiro cenário de testes	p. 47
18	Segundo cenário de testes	p. 49
19	Terceiro cenário de testes	p. 50
20	Quarto cenário de testes	p. 52

21	Módulo XBee	p. 55
22	Esquema de ligação da implementação	p. 59
23	Fluxo de código para o nó fonte	p. 60
24	Fluxo de código para o nó destino	p. 61
25	Fluxo de código para o nó roteador na recepção de pacotes via UART	p. 63
26	Fluxo de código para o nó roteador na recepção de pacotes RREQ e RREP	p. 65
27	Fluxo de código para o nó roteador para tomada das métricas de disponibilidade	p. 66
28	Fluxo de código para o nó roteador na recepção de pacote ERROR	p. 66
29	Vista superior do cenário de testes	p. 67

Lista de Tabelas

1	Resultados da simulação no estudo da métrica	p. 21
2	Configuração do Cenário 1	p. 47
3	Resultados para o Cenário 1	p. 48
4	Configuração do Cenário 2	p. 49
5	Resultados para o Cenário 2	p. 49
6	Configuração do Cenário 3	p. 50
7	Resultados para o Experimento1 do Cenário 3	p. 51
8	Resultados para o Experimento2 do Cenário 3	p. 51
9	Configuração do Cenário 4	p. 52
10	Resultados para o Cenário 4	p. 53
11	Características do ADuC8XX	p. 58
12	Resultados no testes da implementação CA-AODV	p. 68

Lista de Abreviaturas e Símbolos

ACK	Pacote de acknowledge	p. 12
AODV	Ad-hoc On-demand Distance Vector	p. 3
API	Application Programming Interface	p. 56
BID	Broadcast identification number	p. 25
C#	C Sharp	p. 58
CA-AODV	Congestion-Aware AODV	p. 3
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance	p. 16
DSDV	Destination-Sequenced Distance-Vector	p. 12
ERROR	Pacote ERROR	p. 24
HELLO	Pacote HELLO	p. 24
IEEE	Institute of Electrical and Electronics Engineers	p. 2
IETF	Internet Engineering Task Force	p. 18
IP	Internet Protocol	p. 11
ISM	Industrial Scientific and Medical	p. 9
LLC	Logical Link Control	p. 9
LR-WPAN	Low Rate Wireless Personal Area Network	p. 9
MAC	Medium Access Control	p. 8
MANET	Mobile Ad Hoc Network	p. 1
NS-2	Net Simulator	p. 4
PDR	Packet Delivery Ratio	p. 45
PHY	Physical Layer	p. 9
PSR	Estimativa de sucesso no envio	p. 3
QoS	Qualidade de serviço	p. 2
RF	Rádio frequência	p. 56
RREP	Pacote route reply	p. 26
RREQ	Pacote route request	p. 25
RSSF	Redes de sensores sem fio	p. 1
SNR	Relação sinal ruído	p. 3
SSCS	Service Specific Convergence Sublayer	p. 9

UART	Universal Asynchronous Receiver/Transmitter	p. 55
WLAN	Wireless Local Area Network	p. 9
WPAN	Wireless Personal Area Network	p. 9

1 *Introdução*

Uma rede é formada pela coleção de entidades interconectadas promovendo a troca de informação entre si [1]. As redes de sensores sem fio (RSSF) podem ser vistas como um tipo especial de rede *ad hoc* móvel (*Mobile Ad Hoc Network* - MANET). MANETs caracterizam-se como um sistema de nós sem fio dinamicamente auto organizados em termos de: configuração da rede, dos nós roteadores e das estações remotas (*hosts*) [2]. Redes de sensores sem fio são utilizadas para monitorar e, eventualmente, controlar o ambiente no qual se encontram. Esse tipo de rede é tipicamente formada por centenas ou milhares de dispositivos autônômicos chamados nós sensores.

Os nós sensores tendem a ser projetados com pequenas dimensões e para operar remotamente, esses requisitos impõem restrições aos recursos dos nós tais como: energia, capacidade de processamento e armazenamento. Dentre essas limitações a de energia é a principal [3]. Qualquer projeto de RSSF tem no consumo de energia um fator prioritário, busca-se sempre o aumento da vida útil da rede. Tendo em vista o grande número de nós sensores o custo de manutenção da fonte de energia é bastante alto, pois mobilizar e deslocar pessoal para fazer a manutenção de cada nó é uma tarefa extensa e demorada, e portanto bastante onerosa. Às vezes sequer é possível fazer a manutenção da fonte de energia, pois os nós se encontram em regiões onde o acesso é difícil ou mesmo impossível, como é no caso de monitoração da cratera de um vulcão.

Quando a área de cobertura desejada para a rede é grande adota-se um modelo de multi-saltos para a disseminação da informação, pois, devido às métricas de consumo, o alcance dos nós é limitado. A potência requerida para a transmissão aumenta com o quadrado da distância entre a fonte e o destino, ou seja, se a distância entre fonte e destino é R , a potência requerida para a transmissão é proporcional a R^2 . Portanto, múltiplas mensagens em curtas distâncias requerem menor potência que uma única mensagem à longa distância. Se dois nós desejam se comunicar mas estão fora do alcance um do outro, para uni-los deve-se construir uma rota usando nós intermediários e cada pacote de dados deverá executar saltos entre os nós intermediários desde o nó fonte até chegar ao nó de

destino.

O crescente uso de redes sem fio para aplicações de tempo real, tais como voz, vídeo, etc, traz consigo a necessidade de garantias de qualidade de serviço (QoS) em termos de atraso, perdas de pacote e outros parâmetros. Promover QoS em sistemas sem fio é um desafio devido à natureza dinâmica e imprecisa dos estados da rede.

Na vanguarda das tecnologias sem fio existentes está o padrão IEEE 802.15.4 (aprovado em maio de 2003). Esse padrão é voltado para aplicações industriais com baixa taxa de transmissão de dados mas que necessitam de baixa latência na comunicação e baixo consumo de energia.

1.1 **Problemática**

Mesmo para RSSF cujos nós são estacionários, a qualidade do enlace é bastante variável. Essa oscilação na qualidade decorre de fontes de interferência e de modificações nas condições do meio como, por exemplo, a movimentação de obstáculos [4]. As fontes de interferência podem ser externas ou internas, decorrentes de outros enlaces da própria rede.

Dado isto, intensificou-se o número de pesquisas em protocolos de roteamento nos quais as rotas são construídas usando medidas de qualidade dos enlaces. Essas pesquisas trouxeram também a questão de qual parâmetro de qualidade usar e como obtê-lo. Um bom balanceamento desses dois fatores é determinante na escolha do parâmetro, pois um parâmetro que dá uma boa medida de qualidade do enlace pode não ser o mais indicado caso o mecanismo para sua obtenção implique na sobrecarga de pacotes de controle na rede e, portanto, afete o desempenho do protocolo como um todo.

A obtenção dos parâmetros de qualidade pode ser feita por meio de pacotes de controle, específicos para tal fim, ou usando informações já disponíveis no próprio funcionamento da rede, como é o caso do uso de pacotes de dados de enlaces já existentes para obtenção da qualidade das rotas. Cada um desses mecanismos tem suas vantagens e desvantagens.

O uso de pacotes de controle para obtenção da medida de qualidade, como faz Ahmed em [5], provoca aumento da carga de pacotes de controle na rede, além disso esses pacotes são diferentes em termos de tamanho e vazão dos pacotes que realmente usarão a rota, os pacotes de dados. A vantagem do uso de pacotes de controle para obter a medida de qualidade ao invés de usar os enlaces de dados existentes, está na maior capacidade de

análise de diferentes rotas, pois não é necessário que o enlace esteja em uso para que se possa obter sua medida de qualidade. Senel em [6], usando pacotes de controle, obtém uma estimativa de sucesso no envio (PSR) baseado nas características da relação sinal ruído (SNR) no receptor.

Usar pacotes de dados das rotas existentes para obter a qualidade da rota como feito por Karbaschi em [7], Chen em [8] e Yarves em [9], tem como vantagem a não inserção de carga de controle na rede e a compatibilidade entre os pacotes usados para medir a qualidade e os pacotes que realmente usarão a rota. A maior desvantagem do uso desses pacotes está na restrição do universo das rotas de análise, são analisadas apenas as rotas que já estão em uso. Isso não acontece ao usar pacotes de controle, quando é desejada a medida de qualidade de um dado enlace essa medida é obtida esteja esse enlace em uso ou não.

Os métodos existentes para estimação da qualidade das rotas são centrados na análise sobre os enlaces. Para a rede representada na Figura 1, caso sejam usados os próprios pacotes de dados da rota existente ($A \rightarrow B$) para fazer estimativa da qualidade da rota, a medida estará restrita aos enlaces pertencentes à rota ($A \rightarrow C \rightarrow D \rightarrow E \rightarrow B$).

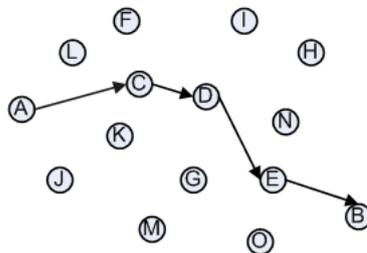


Figura 1: Representação de uma RSSF

Para que alguma outra rota, por exemplo de “J” a “H”, faça uso da informação de qualidade obtida na rota de “A” a “B” é necessário que essa nova rota use algum enlace pertencente à rota $A \rightarrow B$. A aplicabilidade de uma medida de qualidade é restringida ao tomá-la sobre o enlace.

1.2 Objetivos do Trabalho

Com base nas discussões apresentadas anteriormente, o principal objetivo nesta dissertação é desenvolver e avaliar um protocolo de roteamento para redes sem fio usando uma métrica de qualidade focada na análise dos nós da rede, CA-AODV (*Congestion-Aware AODV*). O CA-AODV é baseado no *Ad-hoc On-demand Distance Vector* [10] (AODV),

um protocolo de roteamento bastante difundido no meio acadêmico [11]. A avaliação do CA-AODV é feita comparando-o ao AODV através de simulações para o padrão IEEE 802.15.4, usando como ferramenta o *software* Net Simulator, NS-2 [12] [13].

A métrica utilizada tem como objetivo a busca por rotas menos congestionadas, e que portanto estejam mais propensas a receber novos fluxos de dados.

1.3 Relevância

A aplicação de uma métrica de qualidade sobre um nó, não estando restrita a um dado enlace, aumenta a capacidade de análise dos métodos que usam pacotes de dados para obter a qualidade da rota. No Capítulo 3, será apresentada uma métrica de qualidade para as rotas, que tomada sobre os nós da rede possibilita seu uso para criação de diferentes rotas. Dada a importância que as RSSF vêm ganhando, os resultados apresentados neste trabalho ganham importância, pois, essa nova visão de métrica de análise possibilita tanto a pesquisa de outras métricas similares, voltadas a análise de congestionamento dos nós, como a pesquisa de diferentes usos da métrica proposta neste trabalho.

Há também contribuições nas soluções encontradas no modelo do CA-AODV ao modificar o AODV, acrescentando a esse a funcionalidade de análise das rotas usando métricas de qualidade.

1.4 Estrutura da Dissertação

Esta dissertação está organizada em 7 capítulos, organizados da seguinte forma:

- No Capítulo 2 são apresentados e discutidos os princípios básicos das redes de sensores sem fio. Em sequência é dada uma visão básica sobre o padrão IEEE 802.15.4, sua arquitetura e características de funcionamento. Neste capítulo também é discutido o roteamento em RSSF, foco neste trabalho, e são apresentados algumas das principais dificuldades no gerenciamento e projeto dos protocolos de roteamento em RSSF;
- No Capítulo 3 são apresentadas e discutidas as métricas introduzidas ao modelo do AODV para seleção de rotas. Também são apresentados resultados de simulação desenvolvidos para avaliação da métrica proposta;

-
- No Capítulo 4 são dados detalhes de funcionamento do protocolo AODV, que foi usado como base no desenvolvimento deste trabalho;
 - No Capítulo 5 são dados detalhes do protocolo desenvolvido (CA-AODV), empregando como métrica de seleção de rotas a disponibilidade de recursos relacionados aos nós nas rotas;
 - No Capítulo 6 são apresentados e discutidos os resultados de testes por simulação do CA-AODV, verificando em diferentes cenários a eficiência do CA-AODV quando comparado ao AODV;
 - No Capítulo 7 são apresentados detalhes da implementação do CA-AODV numa plataforma real, assim como, são mostrados resultados de testes efetuados no sistema concebido.

2 Redes de Sensores Sem Fio (RSSF)

Os avanços na eletrônica de baixo consumo e da integração de circuitos levaram ao desenvolvimento de microsensores inteligentes. Tais sensores possuem em geral capacidade de processamento e capacidade de transmissão de dados [14]. As RSSF são a disposição de um grande número de sensores, formando uma rede colaborativa para monitoração de fenômenos em um dado ambiente. A chegada e processamento de dados dos sensores revela a ocorrência de eventos ou propriedades de objetos que estão nas proximidades dos sensores.

Apesar dos nós possuírem individualmente pouca capacidade computacional e capacidade de energia, um esforço colaborativo entre os mesmos permite a realização de uma tarefa maior.

A redução de tamanho e custo dos sensores intensificou o interesse em redes construídas pela disposição aleatória de um grande conjunto de nós sensores para monitoração colaborativa de um ambiente.

Especula-se que o possível uso de uma estrutura colaborativa de nós sensores dispostos de maneira aleatória tenha grande impacto em aplicações tanto de uso militar como civil, como por exemplo, a vigilância de uma zona de combate e monitoração de desastres. Nas aplicações militares a substituição de equipes humanas por nós sensores nas atividades de reconhecimento de terreno inimigo pode ter impacto significativo nas taxas de perda de pessoal. Em caso de desastres de larga escala, nós sensores poderiam ser jogados de um helicóptero assistindo o resgate de vítimas e identificando regiões de risco. Tais ferramentas não só possibilitam um trabalho de resgate mais eficiente como também prezam pela integridade das equipes de socorro.

Contudo, as restrições características dos nós sensores (energia, processamento e armazenamento) combinado ao grande número de nós, trazem grandes desafios no projeto

e gerenciamento da rede. Tais desafios motivam pesquisas de processamento e fusão de dados, coordenação e gerenciamento do sensoriamento e métodos de difusão dos dados.

Para alcançar um sistema energeticamente eficiente cada camada do protocolo tem sua própria política de gerenciamento da energia. Em relação à camada física, em geral, usa-se a técnica *voltage scaling* (regulação da tensão de alimentação afetando a frequência do processador) e alteração do ciclo de trabalho do processador [15]. Para a camada de rede a maioria dos esforços estão no desenvolvimento de protocolos de roteamento mais eficientes no consumo de energia [16].

Os projetos tanto da arquitetura da rede como do sistema de roteamento passam primeiro pela definição de fatores característicos da aplicação, pois diferentes considerações de construção e métricas de desempenho são adotadas a partir do detalhamento da aplicação. A seguir são apresentados alguns parâmetros importantes para a concepção do sistema:

- **Dinâmica da rede:** as RSSF possuem dois componentes básicos: os nós, sensores ou sorvedouros de dados, e os fenômenos monitorados. Quanto aos nós, esse parâmetro especifica sua mobilidade. Em relação aos fenômenos, dependendo da aplicação esses podem ser dinâmicos ou estáticos. Por exemplo, no rastreamento de um objeto a informação de localização é constantemente alterada, o fenômeno é dinâmico, enquanto que em um sistema de prevenção a incêndio o fenômeno é estático. Fenômenos dinâmicos em geral necessitam de comunicação periódica o que implica no aumento de carga na rede;
- **Arranjo dos nós:** os nós podem ser alocados de forma previamente determinada ou de maneira aleatória. A alocação dos nós tem influência direta nos parâmetros de consumo de energia, pois a rota entre os nós sensores e sorvedouros é primordialmente dependente da localização desses. Se a distribuição dos nós na rede não é uniforme, as rotas tendem a se concentrar em um único conjunto de nós nas regiões de menor densidade de nós. Isto significa que os nós, nos quais se concentram as rotas, são mais utilizados e portanto têm sua fonte de energia consumida mais rapidamente;
- **Modelo de envio de dados:** o modelo de envio pode ser contínuo, dirigido a eventos, por solicitação ou híbrido. Esse conceito pode ser confundido com a dinâmica dos fenômenos monitorados, o modelo de envio diz respeito à estratégia de envio de

dados uma vez que o fenômeno monitorado ou um conjunto desses seja computado, independente de como seja a dinâmica do próprio fenômeno. Usando o exemplo anterior de rastreamento de um objeto, um fenômeno dinâmico, o conjunto de informações de posicionamento pode ser disseminado continuamente a cada amostragem ou apenas quando for detectado o evento de passagem do objeto rastreado por um ponto específico. Estratégias híbridas são bastante utilizadas, pode-se ter um conjunto de dados continuamente enviado por demanda. Esse parâmetro tem forte influência no consumo de energia da rede, pois em um nó o rádio é o equipamento com maior consumo de energia;

- **Capacidade dos nós:** em uma rede os nós podem assumir diferentes funções, mesmo que todos apresentem iguais capacidade de processamento, memória e etc. Por exemplo, os nós podem servir puramente para sensoriamento, roteamento ou ter a função de agregação de dados redundantes. A função exercida pelo nó também tem grande impacto no seu consumo.

Redes de sensores sem fio são bastante úteis em contextos onde métodos estatísticos são aplicáveis. Uma vez que servem perfeitamente para coletar muitas amostras sob diferentes circunstâncias de tempo, localidade, e outros fatores.

2.1 Controle de Acesso ao Meio em Redes Sem Fio

O uso de canais compartilhados é a base da comunicação sem fio. Num canal compartilhado entre múltiplos usuários, a recepção de dados pode ser feita por qualquer um dos usuários que esteja no alcance do transmissor. Isto motiva o desenvolvimento de mecanismos de controle do acesso ao meio, que minimizem a ocorrência de colisões nas transmissões. A camada MAC (*Medium Access Control*) define como o meio sem fio de acesso distribuído é partilhado entre os diversos nós da rede.

Os protocolos MAC devem levar em consideração uma situação característica das redes sem fio: o problema dos nós “escondidos” [1]. Essa situação ocorre quando dois nós, fora do alcance um do outro, tentam ao mesmo tempo enviar dados a um mesmo nó receptor.

O problema de coordenação do acesso ao meio em RSSF é mais pronunciado que em outros tipos de redes. Além dos objetivos convencionais, minimizar colisões e otimizar o uso do canal, o protocolo MAC deve ser otimizado quanto ao consumo de energia.

Dentre os diversos protocolos MAC propostos para RSSF, o IEEE 802.15.4 [17] é o que melhor se adapta às necessidades únicas da comunicação sem fio para aplicações industriais, entre dispositivos de controle e sensoriamento. Esse novo padrão foi desenvolvido visando aplicações para redes pessoais sem fio com baixa taxa de dados (*Low Rate Wireless Personal Area Network* – LR-WPAN).

WPANs são redes sem fio usadas para transportar informação sobre distâncias relativamente curtas. Diferentemente das redes locais sem fio (*Wireless Local Area Network* - WLAN), as conexões efetuadas via WPANs envolvem pouca ou nenhuma infra-estrutura. Isto permite que sejam implementadas soluções de baixo custo e de alta eficiência no consumo de energia.

Algumas das características mais desejáveis numa LR-WPAN são a facilidade de instalação, a transferência confiável de dados, operação de curto alcance, o baixo custo e uma significativa vida útil da bateria, enquanto mantém do ponto de vista do usuário um protocolo MAC simples e flexível.

O IEEE 802.15.4 define as camadas MAC e PHY (*Physical Layer*), utilizando a banda de ISM (*Industrial Scientific and Medical*) que não requer licenciamento. Os rádios operam nas faixas de frequência de 868 *MHz* na Europa, 915 *MHz* nos Estados Unidos e 2,4 *GHz* nos outros lugares do mundo, com taxas de transmissão de 20 *Kbps*, 40 *Kbps* e 250 *Kbps* respectivamente. Os dispositivos usam um endereçamento fixo de 64 *bits* e um configurável de 16 *bits*, uma única rede pode acomodar até 2^{16} dispositivos.

Na Figura 2 apresenta-se uma representação da pilha do IEEE 802.15.4, as camadas superiores (rede, aplicação, etc.) podem acessar a camada MAC diretamente via as primitivas que essa disponibiliza ou de maneira indireta via as camadas 802.2 LLC (*Logical Link Control*) e SSCS (*Service Specific Convergence Sublayer*). No IEEE 802.15.4 a camada de enlace foi quebrada em 802.2 LLC e MAC [18]. A sub-camada 802.2 LLC estabelece e mantém a comunicação com outros dispositivos e fornece conectividade com servidores quando os dados estão sendo transferidos. A sub-camada SSCS serve de interface entre a 802.2 LLC e MAC além de definir a topologia de rede a ser usada [19].

2.2 Roteamento em RSSF

O mecanismo de roteamento consiste nas tarefas de descoberta de rotas e sua manutenção. Na manutenção estão incluídos mecanismos de monitoração e reconstrução da rota, uma vez que essa seja sinalizada como inoperante.

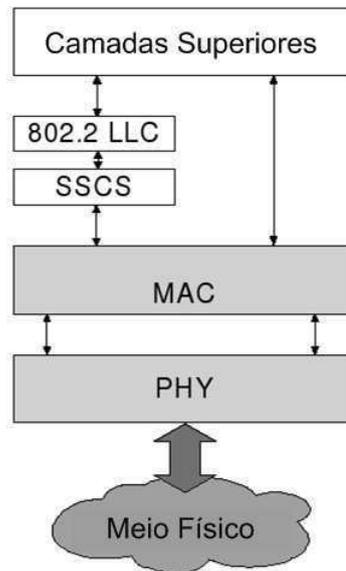


Figura 2: Pilha IEEE 802.15.4

Há uma corrente de autores que classificam os protocolos quanto ao seu modo de operação como proativos, reativos ou híbridos [20].

Nos protocolos proativos as rotas são construídas e mantidas para todos os nós, mesmo que esses não participem da comunicação. Tais métodos reagem a mudanças na topologia mesmo que essas mudanças não afetem o tráfego.

Os protocolos reativos são baseados num comportamento por demanda. As rotas são determinadas apenas quando necessárias. Esses protocolos podem reduzir significativamente a carga de pacotes na rede quando o tráfego é relativamente baixo e as mudanças na topologia não são muito frequentes e drásticas.

Protocolos híbridos combinam conceitos proativos e reativos sem carregar a rede com pacotes de controle.

O roteamento em RSSF possui fatores inerentes a sua natureza que dificultam seu gerenciamento, como as possíveis mudanças nas condições do meio e na topologia da rede. Há grande dificuldade em garantir desempenho na transmissão de dados em tempo real ante ao imprevisível comportamento da qualidade do enlace.

Alguns protocolos usados em RSSF são adaptações de protocolos existentes para outros tipos de rede e por isso em geral usam o número mínimo de saltos como critério único na seleção das rotas [21]. Contudo, essa métrica, o número mínimo de saltos, comumente resulta na seleção de rotas com um desempenho muito baixo em relação a escolha

ótima, em termos de perdas de pacotes. Essas rotas possuem uma taxa de perda de pacotes baixa suficiente para serem aceitas pelo protocolo mas alta suficiente para consumir a capacidade do enlace devido a retransmissões [22].

O roteamento em RSSF apresenta complexidade no seu projeto, devido a algumas características, que as distinguem das redes convencionais. Em primeiro lugar devido ao grande número de nós não é possível construir um esquema global de endereçamento, assim os mecanismos baseados em endereço IP não são aplicáveis. Segundo, o fluxo de dados é em geral feito de muitos nós sensores a um particular ponto (sorvedouro), além disso há redundância nos dados gerados e essa redundância pode ser explorada pelos mecanismos de roteamento.

Dadas as características únicas de cada rede os protocolos de roteamento geralmente são dedicados, implementados para satisfazer a apenas um tipo de rede. Na busca pelo uso racional de energia, diferentes estratégias são adotadas nas redes dirigidas a eventos [23], nos protocolos que utilizam informações geográficas dos nós sensores [24], entre outras.

Atualmente uma estratégia bastante usada para conseguir algum ganho de desempenho é promover interação entre os módulos do dispositivo, seja essa interação entre as camadas do protocolo, entre o protocolo de roteamento e a aplicação ou entre o protocolo de roteamento e o *hardware*.

2.3 Otimização da Pilha de Protocolos

Os sistemas de comunicação têm em geral sua arquitetura baseada em camadas formando uma pilha de protocolos. A modularização do protocolo numa pilha facilita a construção de aplicações e lhe confere portabilidade. Enquanto esses protocolos atingem um alto desempenho, em termos dos parâmetros relacionados com cada camada, suas camadas não são conjuntamente concebidas para maximizar o desempenho da rede como um todo enquanto o consumo de energia é minimizado [25].

Recentemente foram propostos diferentes paradigmas para os protocolos de roteamento de forma que as redes satisfaçam certos requisitos de QoS. Uma possibilidade é o uso de um modelo *cross-layer* na pilha de protocolos. Numa abordagem *cross-layer* as diferentes camadas do protocolo interagem entre si, seja pelo compartilhamento de dados ou por intervenção nas tarefas das camadas inferiores pelas superiores [25]. O ganho de desempenho vem com o custo da perda de modularidade da pilha de protocolos.

Um projeto *cross-layer* pode conduzir a uma arquitetura espaguete, sem estrutura lógica, o que inibe a evolução dos sistemas e dificulta sua manutenção [26]. As considerações sobre a arquitetura podem ser críticas para sua proliferação maciça, quando não há comprometimento com a arquitetura a cada atualização do sistema é necessária uma completa re-estruturação do modelo. Contudo, a possibilidade de otimização do desempenho por meio de uma arquitetura *cross-layer* não pode ser ignorada, mas seu uso deve ser feito com bastante precaução.

Independente do método de projeto usado, a busca por QoS no roteamento é em geral baseada em estimativas sobre o estado do enlace [27]. Para estimar esse estado são usadas diferentes informações tais como atrasos, taxa de perdas e de erro, etc. A qualidade da informação usada na definição da métrica de roteamento determina o ganho de desempenho [26].

Karbaschi [7], aplicando uma abordagem *cross-layer*, obtém a qualidade do enlace baseado no histórico de falhas de recebimento do pacote de reconhecimento (*Acknowledge - ACK*). O autor propõe uma extensão do protocolo de roteamento *Destination-Sequenced Distance-Vector* (DSDV) para que esse dê suporte a QoS. Essa extensão do DSDV faz uso do parâmetro de qualidade para classificar os enlaces e montar suas rotas. Karbaschi obtém sua métrica de qualidade usando o próprio procedimento de envio de dados sem inserir tráfego na rede por pacotes de prova para fazer tal medição, diminuindo a inserção de carga extra na rede.

Tian [28] propõe um novo protocolo para o IEEE 802.11 que, assim como apresentado por Karbaschi em [7], infere a qualidade do enlace. Seu fator de qualidade é dado pelo tempo decorrido da transmissão do pacote até o recebimento do pacote de ACK. Nesse parâmetro é presumido conter informações de congestionamento do meio e disponibilidade de recursos do transmissor e receptor.

Alguns protocolos fazem uso de diversas informações de diversas camadas (MAC, PHY,...), buscando de cada camada informações com características singulares de forma a detectar diferentes possíveis causas dos problemas de perdas de pacotes, interferência, etc. Ahmed em [29] combina parâmetros como atraso de recepção dos pacotes por parte do receptor, medidas da relação sinal-ruído e energia remanescente no receptor para composição das rotas, dessa forma busca-se proteção contra diferentes causas de perdas de pacote e falhas na rede.

Os protocolos apresentados nesses trabalhos usam diferentes informações (contagem de falhas na transmissão, tempo de resposta, medidas de relação sinal-ruído, etc) para

obter medidas de qualidade dos enlaces, sejam essas medidas obtidas por pacotes de controle ou pacotes de dados.

2.4 Conclusão

O presente capítulo apresentou os principais aspectos envolvidos em RSSF, incluindo o funcionamento básico, exemplos de aplicação, os requisitos comuns de funcionamento, arquitetura da rede e suas características. Dentre as características apresentadas, destaca-se a limitação das redes quanto à fonte de energia dos nós, configurando o gerenciamento desse fator como requisito primário no projeto das redes. As RSSF são soluções dedicadas à aplicação, ou seja, dependendo de fatores característicos da aplicação são adotadas diferentes considerações de projeto e arquitetura da rede.

Também foi apresentada uma discussão básica sobre o padrão IEEE 802.15.4. Esse padrão foi desenvolvido visando principalmente a aplicações industriais sendo, portanto, focado neste trabalho. Uma característica importante do IEEE 802.15.4 é o seu baixo consumo, isso faz com que esse padrão seja ideal às RSSF com baixas taxas de transmissão.

Os protocolos de roteamento são também foco neste capítulo, onde é apresentada uma classificação, comumente usada, dos protocolos de roteamento quanto ao seu modo de operação: proativos, reativos ou híbridos. Nos protocolos proativos as informações sobre as rotas são montadas e atualizadas mesmo que essas informações não sejam necessárias. Os protocolos reativos trabalham sob demanda, só são ativados quando há necessidade de usar alguma rota. Os híbridos combinam características desses dois tipos de protocolos.

As RSSF, devido a sua natureza, possuem características que dificultam a tarefa de roteamento: mudanças nas condições do meio e na topologia da rede. Buscando melhorar algum parâmetro de desempenho das redes, estão sendo continuamente pesquisados protocolos que empregam alguma medida de qualidade na construção das rotas. A obtenção e uso dessa medida é muitas vezes combinada ao emprego de um modelo *cross layer* na pilha de protocolos. Nos modelos *cross layer* são empregadas interações entre as diferentes camadas da pilha de protocolos, quebrando assim a modularidade da pilha.

Os trabalhos relacionados com o emprego de alguma métrica de qualidade, usam medidas de qualidade tomadas sob o ponto de vista do enlace, ou seja, diferentes informações são usadas para obter algum parâmetro relacionado à qualidade dos enlaces e, então, construir uma medida de qualidade da rota.

3 *Estudo da Métrica de Qualidade Aplicada no Roteamento em RSSF*

Neste trabalho propõe-se o estudo de dois parâmetros relacionados à qualidade das rotas e sua utilização no desenvolvimento de um protocolo de roteamento para RSSF.

Neste capítulo, as métricas de qualidade e o protocolo propostos são discutidos, e é apresentado um exemplo ilustrativo de dependência da métrica em relação ao congestionamento da rota.

3.1 Métrica de Qualidade

Em [28], [7] e [5] a seleção das rotas é feita baseada numa medida de qualidade dos enlaces, obtida por diferentes mecanismos. Neste trabalho propõe-se montar as rotas a partir de uma medida de qualidade, só que, diferentemente dos trabalhos anteriores, tomada sobre a disponibilidade dos nós e não mais sobre os enlaces.

Uma rota é formada por enlaces, mas também pode ser vista como uma seleção de nós. Para a rede sem fio representada na Figura 3, a rota que vai do nó “A” para o nó “B” é formada pelos seguintes enlaces: $A \rightarrow C$, $C \rightarrow D$, $D \rightarrow E$ e $E \rightarrow B$. Uma outra forma de visualizar isto é que a rota desde sua origem até seu destino percorre os nós: A, C, D, E e B.

O ganho nessa troca de pensamento sobre a montagem da rota vem da possibilidade de utilização da informação de qualidade de uma rota já existente na construção de uma outra. Se uma métrica de qualidade diz respeito, por exemplo, ao enlace $C \rightarrow D$, não seria possível utilizar essa métrica durante a construção de uma rota que usasse o enlace $C \rightarrow F$, é necessário obter a medida (qualidade de $C \rightarrow F$) agora que ela é necessária. No entanto, caso tenha-se em mãos uma medida relacionada diretamente com o nó “C”, sua

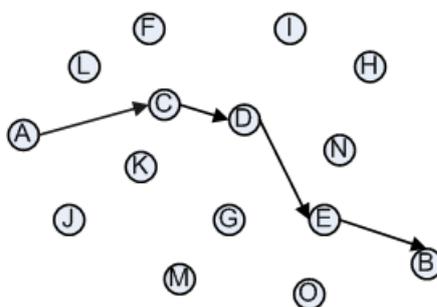


Figura 3: Diagrama de uma rede *ad hoc* sem fio com uma rota

disponibilidade e do meio no qual esse se encontra, admite-se o uso dessa medida para a construção de diferentes rotas.

Para ilustrar essa afirmação, toma-se a rede representada na Figura 4. Há duas rotas de “A” a “B” e de “J” a “H”, ambas em pleno uso, com tráfego de dados. Se uma terceira rota fosse necessária, por exemplo de “L” a “O”, provavelmente alguns nós que já estão em uso nas rotas $A \rightarrow B$ e $J \rightarrow H$ seriam usados na formação de $L \rightarrow O$, e sobre esses nós já em uso é possível tomar uma medida de sua disponibilidade para fazer a construção da rota de $L \rightarrow O$. Para esse cenário é possível, por exemplo, selecionar entre os nós “C” e “K” aquele que está mais “disponível” para a construção de uma nova rota. É argumentado que a melhor rota a ser selecionada é aquela cujos nós possuem maior disponibilidade de recursos, *hardware* e o meio no qual o nó se encontra.

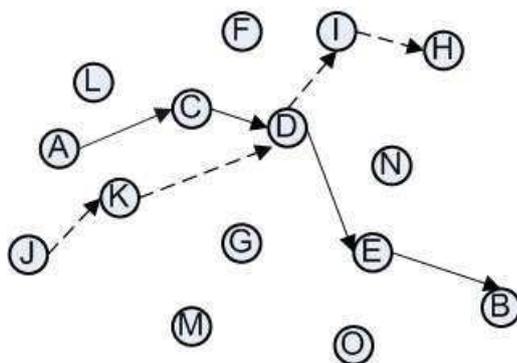


Figura 4: Diagrama de uma *ad hoc* sem fio com duas rotas

Para a utilização de uma métrica sobre a disponibilidade dos nós, como foi a pouco descrito, é necessário que apenas fatores referentes a recursos do nó em questão influenciem na medida, sejam esses recursos de *hardware* ou relacionados ao meio no qual o nó se encontra (seu domínio de colisão).

Um parâmetro que atinge esses requisitos, podendo ser usado como métrica para

análise dos recursos dos nós, é o atraso do recebimento do pacote de ACK. Esse atraso é o intervalo de tempo entre o envio do pacote de dados pela camada de aplicação no nó fonte até o recebimento do pacote de ACK enviado pelo nó destino. Utilizando esse parâmetro é possível inferir tempos relacionados à carga do nó (preenchimento da pilha) e nível de contenção do meio. O tempo de recebimento de ACK (t_{ack}) pode ser descrito pela seguinte equação:

$$t_{ack} = t_{proc}^f + t_{csma/ca}^f + t_{prop}^f + t_{proc}^d + t_{csma/ca}^d + t_{prop}^d \quad (3.1)$$

onde:

- t_{proc}^f é o tempo de processamento do pacote de dados pelo nó fonte. Esse tempo é proporcional ao nível da pilha do nó fonte e outros mecanismos de processamento do pacote nas camadas inferiores relacionados com o tamanho do pacote.
- $t_{csma/ca}^f$ é o tempo gasto pelo mecanismo CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*) de acesso ao meio do nó fonte. Esse tempo é proporcional ao nível de contenção do domínio de colisão do nó fonte.
- t_{prop}^f é o tempo de propagação do pacote de dados desde o nó fonte até o nó destino. Como se trata de uma onda eletromagnética propagando no ar esse tempo pode ser desconsiderado.
- t_{proc}^d é o tempo de pré-processamento do pacote de dados no destino para geração do pacote de ACK. Esse tempo é proporcional ao tamanho do pacote e pode ser considerado constante para pacotes de tamanho fixo, k para a Equação 3.2. Após o pré-processamento o pacote é colocado numa pilha onde fica a espera da recepção propriamente dita.
- $t_{csma/ca}^d$ é o tempo gasto pelo mecanismo CSMA/CA do nó destino. No entanto, para o padrão IEEE 802.15.4 não há mecanismo de CSMA/CA para pacotes de ACK, portanto esse tempo é nulo.
- t_{prop}^d é o tempo de propagação do pacote de ACK do nó destino até o nó fonte, também pode ser desconsiderado.

Assim, a Equação 3.1 pode ser reescrita como:

$$t_{ack} = t_{proc}^f + t_{csma/ca}^f + k \quad (3.2)$$

Observando a Equação 3.2, conclui-se que o tempo de recebimento do pacote de ACK pode ser usado como métrica de disponibilidade de recurso de um dado nó.

No padrão IEEE 802.15.4, a obtenção da métrica de disponibilidade (atraso do pacote de ACK) usa as duas primitivas para o envio de pacotes *unicast*: MCPS-DATA.request e MCPS-DATA.confirm. A primeira é uma mensagem de requisição de envio *unicast* de um pacote de dados (sentido: camadas superiores \rightarrow MAC, vide Figura 2) e a segunda é a notificação de recebimento do pacote de ACK (sentido: MAC \rightarrow camadas superiores, vide Figura 2). Assim para o IEEE 802.15.4 a medida de atraso do pacote de ACK é relativamente simples de ser obtida, sem necessidade de usar uma abordagem *cross-layer* e mantendo a modularidade da pilha.

A outra métrica, também aplicada neste trabalho na análise dos nós e seleção de rotas, é a taxa de uso dos nós. Esse parâmetro é obtido para os nós que já estão em funcionamento na rede, e que portanto estão recebendo pacotes de dados de diferentes fontes a uma dada taxa. Cada nó que esteja roteando dados para alguma rota mantém informações sobre os tempos de chegada dos pacotes, ao receber o pacote de requisição de nova rota sua taxa de recebimento de pacotes de dados é calculada e acrescentada ao pacote de requisição de rota para repasse. Esse parâmetro também atende o requisito de referir-se apenas ao nó de análise.

Para a composição das métricas de disponibilidade, o atraso de ACK e a taxa de uso dos nós, são usadas informações das rotas em uso na rede.

3.2 Protocolo Proposto

No contexto deste trabalho propõe-se o desenvolvimento de um protocolo de roteamento reativo para RSSF, usando o padrão IEEE 802.15.4. Segundo Broch [30] e Johansson [31], protocolos de roteamento reativos são mais adaptáveis a sistemas *ad hoc*.

O protocolo desenvolvido (CA-AODV) é uma modificação do AODV, aliado à métrica de atraso do pacote de ACK e à taxa de uso dos nós para obter uma medida de disponibilidade dos nós na rota. A escolha do AODV vem por esse ser um dos mais estudados protocolos de roteamento reativo para RSSF [32], dada sua eficiência em redes *ad hoc*. O

AODV é especificado em RFC3561 pelo IETF (*Internet Engineering Task Force*).

O CA-AODV é voltado para aplicações em RSSF de nós estáticos e enlaces simétricos, ou seja, os nós não possuem qualquer mobilidade e a velocidade de comunicação de um dado enlace é a mesma não importa o seu sentido. Esse novo protocolo mantém a maioria dos mecanismos do AODV, salvo os procedimentos de “construção” das rotas. O novo mecanismo de “construção” faz uso de duas métricas para seleção das rotas: a) taxa de uso dos nós; e b) atraso do recebimento do pacote de ACK. O AODV utiliza uma única métrica para a construção das rotas: o tempo de chegada da requisição de nova rota.

A taxa de uso é obtida pela recepção de pacotes de dados e o atraso de ACK é obtido na transmissão *unicast* de pacotes, portanto essas são medidas tomadas sobre os nós que já estão em funcionamento por alguma rota, estão recebendo e enviando pacotes de dados. A informação de disponibilidade da rota é composta por três parâmetros:

- a métrica de atraso de ACK acumulada da rota, ou seja, cada nó ao longo da rota que possua informação sobre sua medida de disponibilidade acrescenta-a ao pacote de descobrimento de rotas;
- a taxa de uso acumulada dos nós ao longo rota, ou seja, cada nó que possua informação sobre sua taxa de uso acrescenta-a ao pacote de descobrimento de rotas;
- quantidade de nós ao longo da rota que já estão em uso e que, portanto, possuem métricas para análise de disponibilidade da rota.

Esses três parâmetros são tratados para obter a medida de qualidade da rota.

Voltando ao exemplo da Figura 4 para formação da rota de “L” a “O”, o nó “G” usando a métrica de disponibilidade de cada rota julgará entre os nós “C” e “K” aquele mais susceptível a uma nova rota. O nó “G” recebe informação de duas rotas distintas, uma passando por “C” e outra por “K”. Esses dois nós possuem seus parâmetros de análise, e essa informação é repassada até chegar a “G” que baseado nos três parâmetros de análise de cada rota toma decisão de qual dessas usar.

Foi preestabelecida a busca pela minimização de carga a ser acrescentada pelas modificações no modelo original, ou seja, o novo protocolo deve primar pela não inclusão de mecanismos que signifiquem em acréscimo de pacotes de controle, tudo isso sem a perda de características importantes do AODV como a inexistência de *loops*. A modificação será feita apenas no mecanismo de construção de rotas do AODV.

3.3 Método Estatístico para Análise dos Resultados

Os resultados apresentados neste trabalho são baseados na análise estatística dos resultados de diversos testes independentes de um mesmo cenário de experimento. A ideia básica é que uma determinada conclusão não pode abranger todas as características do sistema. Deve-se obter uma medida probabilística do intervalo no qual estão contidas as características do sistema [33].

Na maioria dos problemas do mundo real, as características de uma dada população¹ (média, por exemplo) é desconhecida, e a intenção é estimar essas características a partir de uma dada quantidade de experimentos, uma vez que seu valor real só poderia ser obtido pela repetição infinita do experimento.

O método de estimação dos parâmetros aplicado baseia-se no modelo de análise da distribuição z apresentado por Jain [33], considerando um nível de confiança $\rho = 95\%$ e um nível de significância ou erro $\alpha = 5\%$. Esse método é aplicável somente para um conjunto de amostras maior ou igual a 30 amostras [33].

O método consiste na determinação do intervalo de confiança, dado pelos limites (c_1 e c_2), no qual o parâmetro desejado, por exemplo, a média μ se encontra com probabilidade de $1 - \alpha$.

Usando o teorema do limite central [33], é possível determinar a distribuição da média das amostras. Esse teorema estabelece que se um conjunto finito de n observações, $\{x_1, x_2, \dots, x_n\}$, são independentes e foram tomadas de uma mesma população possuindo média μ e desvio padrão σ , então média (\bar{x}) para um grande conjunto de amostras tende a uma distribuição normal com média μ e desvio padrão σ/\sqrt{n} :

$$\bar{x} \approx N\left(\mu, \frac{\sigma}{\sqrt{n}}\right) \quad (3.3)$$

Da relação 3.3 é obtido o intervalo de confiança:

$$\left(\mu - z_{1-\alpha/2} \times \frac{\sigma}{\sqrt{n}}, \mu + z_{1-\alpha/2} \times \frac{\sigma}{\sqrt{n}}\right) \quad (3.4)$$

onde, $z_{1-\alpha/2}$ é igual a 1,96 para $1 - \alpha = 95\%$, definido na Tabela A.2, do Apêndice A, da referência [33].

¹Coleção de unidades individuais, que podem ser pessoas ou resultados experimentais, com uma ou mais características comuns, que se pretendem estudar.

Então, baseado num conjunto reduzido de n amostras, $\{x_1, x_2, \dots, x_n\}$, tomadas por n experimentos independentes, é possível determinar o intervalo de confiança, (c_1, c_2) , no qual a amostra estará contida em 95% dos casos, quando o conjunto de amostras é expandido.

O intervalo de confiança depende da quantidade de amostras, n , usada para sua determinação. Quanto maior n mais confiável é a medida, contudo, maior também é o esforço na obtenção dos resultados. É preciso determinar o menor n , n_{min} , no qual não há perda de precisão ou confiabilidade.

Para $\alpha = 5\%$, tem-se o intervalo de confiança dado por:

$$(\mu(1 - 0,05), \mu(1 + 0,05)) \quad (3.5)$$

Combinando as Expressões 3.4 e 3.5, é obtida a quantidade de experimentos necessária para garantir a confiança requerida ($\rho = 95\%$), dada por:

$$n_{min} = \left(\frac{1,96 \times \sigma}{0,05 \times \mu} \right)^2 \quad (3.6)$$

3.4 Simulações

Usando o NS-2, por meio de simulação, foram feitos testes para verificar a influência do congestionamento no meio sobre o atraso na recepção do pacote ACK. O cenário usado é visto na Figura 17, nesse cada nó está no alcance dos outros dois, ou seja, todos estão sob o mesmo domínio de colisão.

O atraso do pacote ACK foi tomado sob o ponto de vista do nó “1” com o nó “0” como sorvedouro, ou seja, para o nó “1” as transmissões do nó “2” ao nó “0” são interferência no meio. Para cada experimento, tanto o nó “1” como o nó “2” podem fazer transferência de dados para o nó “0” a uma taxa constante. Foram realizados testes com duas taxas de transferência 1,16 *kbps* e 11,6 *kbps* para o nó “2” e uma única de 8,8 *kbps* para o nó “1”.

Os testes consistiram em três experimentos, no primeiro (Exp1) somente há comunicação do nó “1” (8,8 *kbps*) com o nó “0”, no segundo experimento (Exp2) tanto o nó “1” (8,8 *kbps*) como o nó “2” (1,16 *kbps*) transmitem para o nó “0” e no último (Exp3) os nós “1” (8,8 *kbps*) e “2” (11,6 *kbps*) transmitem para o nó “0”. Agora tem-se em mãos três situações distintas do ponto de vista do nó “1”, a primeira é um caso em que não há

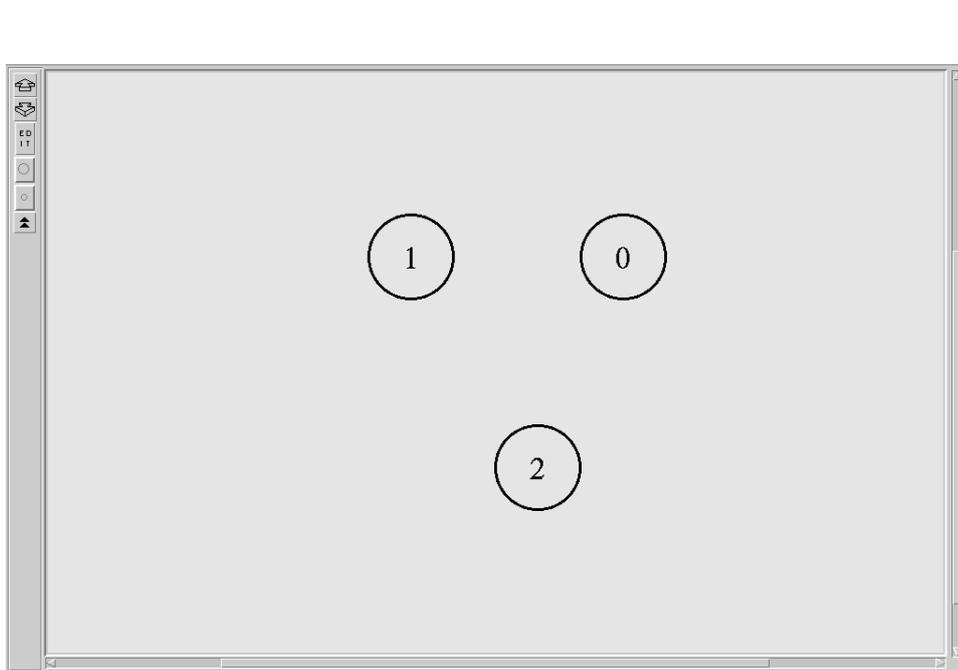


Figura 5: Cenário de avaliação da métrica

interferência e portanto o acesso ao meio é mais rápido e nas outras duas temos diferentes níveis de contenção do meio.

Cada experimento foi repetido 31 vezes usando diferentes sementes (*seeds*), para que esses sejam independentes. Na obtenção das sementes foi usado o gerador de números aleatórios existente no NS-2, baseado no gerador recursivo múltiplo (*Multiple Recursive Generators*) desenvolvido por Pierre L'Ecuyer [34].

Em cada repetição do experimento foi adquirido um conjunto de amostras do atraso de ACK, sobre o qual é calculada a média (x_i). A coleção de médias dos diferentes testes constitui o conjunto de amostras, $\{x_1, x_2, \dots, x_n\}$, usado para obter n_{min} e o intervalo de confiança. Via as Expressões 3.4 e 3.5 foram obtidos os resultados apresentados na Tabela 1.

Tabela 1: Resultados da simulação no estudo da métrica

experimento	média (μ)	intervalo de confiança ($\mu(0, 95) - \mu(1, 05)$)	n_{min}
Exp1	0,003125438505	(0,00296917 - 0,00328171)	0,207756 \approx 1
Exp2	0,003328004935	(0,0031616 - 0,00349441)	2,18277 \approx 3
Exp3	0,004295773960	(0,00408099 - 0,00451056)	153,884 \approx 154

Para todos os resultados, apresentados na Tabela 1, é visível a influência da interferência no meio sobre os atrasos. Quanto à sensibilidade, comparando o atraso médio dos experimentos 2 e 3 observa-se um aumento de 10 vezes no congestionamento (de 1,16 *kbps* para 11,60 *kbps*) significou em aproximadamente 29% de aumento do atraso médio de ACK (de $\approx 0,003328$ s para $\approx 0,004296$ s).

Nas Figuras 6 e 7 são apresentadas as curvas para o atraso de ACK de um único caso, resultado da simulação para uma semente. Nessas figuras, é vista uma variação do atraso de ACK relativamente alta mesmo quando o meio está livre. Essa variação se deve ao algoritmo do mecanismo de CSMA/CA do padrão IEEE 802.15.4, que estabelece um período de espera aleatório antes de realizar a verificação de portadora para acesso ao meio (*Clear Channel Assessment - CCA*), ou seja, mesmo que o meio esteja livre há um intervalo de espera aleatório antes de executar o CCA.

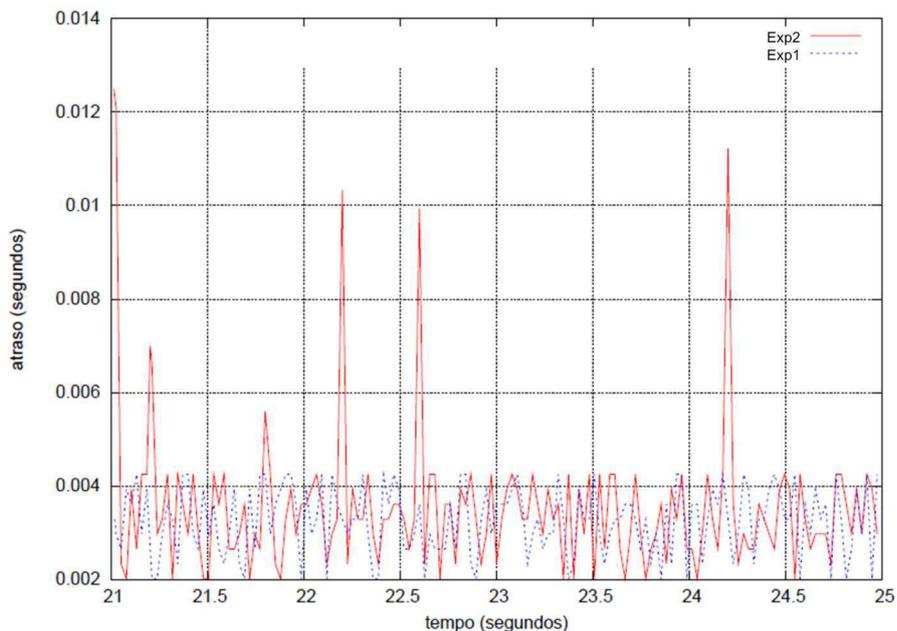


Figura 6: Comparativo curvas do atraso do pacote ACK para Exp1 (sem congestionamento) e Exp2 (com congestionamento)

Esses resultados corroboram as expectativas de uso dos atrasos dos pacotes de ACK na análise de contenção do meio.

3.5 Conclusão

Neste capítulo foram apresentadas e discutidas as duas métricas de qualidade propostas neste trabalho. O uso das métricas propostas se diferencia de outros trabalhos por

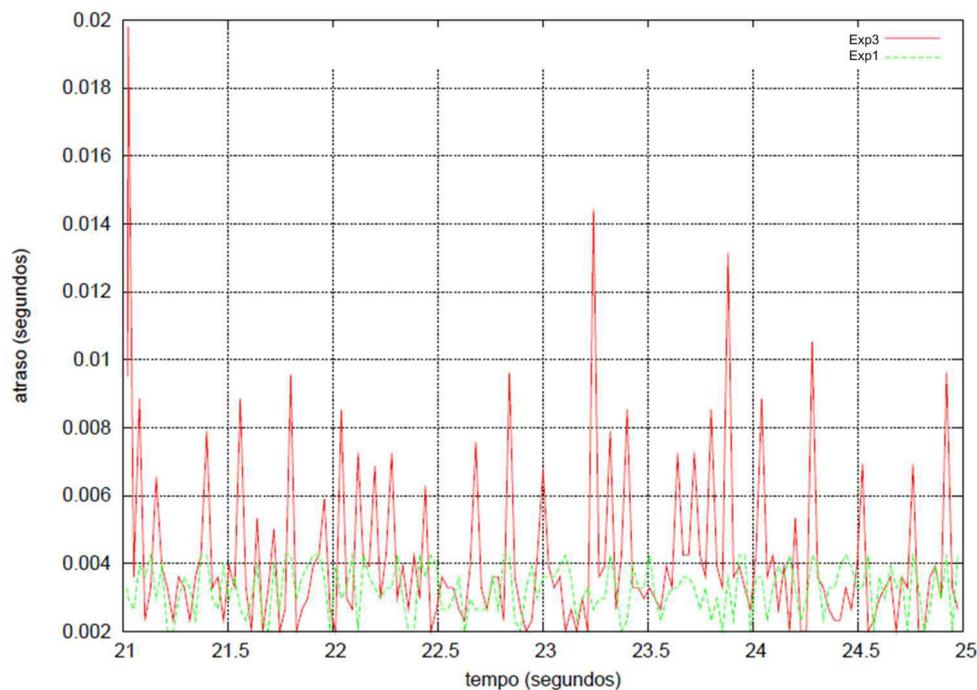


Figura 7: Comparativo curvas do atraso do pacote ACK para Exp1 (sem congestionamento) e Exp3 (sem congestionamento)

serem tomadas sob a perspectiva dos nós e não dos enlaces. Uma métrica de qualidade relacionada ao enlace tem seu uso restringido em comparação a uma métrica tomada sob o ponto de vista dos nós.

Para obter a métrica de qualidade pode-se usar pacotes de controle específicos para tal fim ou os pacotes de dados das rotas já existentes. Usar pacotes de controle permite tomar a medida de qualquer enlace, esteja esse em uso ou não, ao custo de aumentar a carga de pacotes de controle da rede. Usar pacotes de dados das rotas já existentes tem a vantagem de não acrescentar carga à rede, mas tem seu alcance de análise restringido à apenas os enlace já em uso.

Quando a obtenção da métrica usa os pacotes de dados das rotas em uso, empregar uma métrica relacionada com os nós e não somente aos enlaces potencializa sua aplicabilidade.

Os testes de avaliação da dependência do atraso de ACK em relação ao congestionamento do meio, mostrados na Seção 3.4, validam as expectativas de uso dessa métrica na análise da contenção do meio, demonstrado na Equação 3.2. No entanto, para o cenário estudado os resultados mostraram uma baixa sensibilidade do atraso de ACK à variações do meio, para um aumento de 10 vezes no congestionamento verificou-se um acréscimo de 29% ao atraso de ACK.

4 *Ad-Hoc On-demand Distance Vector (AODV)*

Neste capítulo é dada uma breve introdução sobre o AODV [10], que foi a base para a construção do CA-AODV. Uma vez que as etapas de operação do AODV foram conservadas é importante entendê-las melhor.

AODV é um protocolo projetado para uso em redes *ad hoc*, é um protocolo reativo, onde as rotas são construídas apenas quando necessárias. Utiliza tabelas de roteamento, uma entrada para cada destino. O protocolo oferece uma rápida adaptação às condições dinâmicas dos enlaces, baixo processamento e reduzida taxa de pacotes de controle.

O protocolo consiste em dois processos básicos: descoberta e manutenção de rotas [4]. Para tais processos há diferentes tipos de mensagens:

- RREQ (*Route Request*) é um pacote (*broadcast*) de requisição de rota usado sempre que uma nova rota é necessária;
- RREP (*Route Reply*) mensagem (*unicast*) de resposta ao RREQ;
- HELLO são pacotes enviados periodicamente em *broadcast* para checar a presença de nós vizinhos imediatos ativos;
- ERROR é o pacote que sinaliza ao nó fonte a “quebra” de uma rota.

A descoberta de rotas é formada por ciclos busca e resposta. A informação da rota é distribuída em cada um dos nós ao longo do caminho. Na construção de uma rota da fonte até o destino, em cada nó são computadas duas rotas: rota direta e rota reversa. A rota direta tem sentido fonte→destino, enquanto que a rota reversa tem sentido destino→fonte. O AODV emprega vetores de distância no roteamento. O vetor de distância significa o quão longe e em que direção está meu destino. No AODV as informações pertinentes são: a quantidade de saltos e o próximo salto em direção ao destino.

4.1 Descoberta de Rotas

Processo iniciado sempre que um nó fonte deseja enviar dados a um destino cuja informação de rota não possui. Considerando a RSSF da Figura 8, quando o nó “A” requisitar rota até o nó “B”, aquele envia pacotes RREQ em *broadcast* com destino ao nó “B”. Cada nó intermediário ao receber segue os seguintes passos:

- Checa se possui informação de rota até o destino:
 - se possuir envia RREP de volta até o nó fonte;
 - caso não possua:
 - * guarda informação da rota reversa, até o nó fonte:
 1. endereço IP do nó destino;
 2. endereço IP do último salto;
 3. identificador do pacote RREQ (BID);
 4. tempo de expiração da rota reversa;
 5. número de sequência do nó fonte.
 - * re-envia o pacote RREQ *broadcast*.

O BID é incrementado sempre que o nó fonte envia um RREQ, assim os RREQ são identificados de maneira única através do endereço IP do nó fonte e do BID. Se um nó recebe um RREQ com o mesmo par (endereço do nó fonte; BID) esse novo RREQ é descartado. Além do BID há outros dois números de sequência no RREQ: da fonte e do destino. O número de sequência da fonte é usado para manter atualizada informação da rota reversa (até o nó fonte) e o número de sequência do destino especifica o quão recente deve ser a rota até o destino para que essa seja aceita. Na Figura 9 é ilustrada uma representação da estrutura do pacote RREQ.

Na descoberta de rota o pacote RREQ é propagado ao longo da rede até chegar em “B” que processa o pacote e envia o RREP através da rota reversa.

No processo de descoberta são montadas duas rotas, uma até a fonte do RREQ (rota reversa) e outra até o destino do RREQ (rota direta). Para a tabela de roteamento não há distinção entre a rota reversa e a rota direta.

A rota reversa é construída conforme o RREQ é propagado, de forma que ao chegar no destino, o nó “B” na Figura 8, essa já está configurada. Para construir essa rota cada um

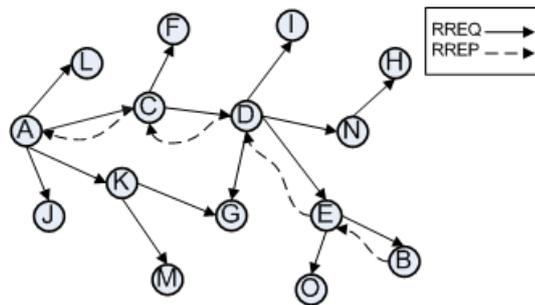


Figura 8: Descoberta de rota

	→ 1 <i>byte</i> , tipo do pacote
	→ 2 <i>bytes</i> , reservado
	→ 1 <i>byte</i> , contagem de saltos
	→ 4 <i>bytes</i> , identificador do RREQ (BID)
	→ 4 <i>bytes</i> , endereço IP do destino
	→ 4 <i>bytes</i> , número de sequência do destino
	→ 4 <i>bytes</i> , endereço IP da fonte
	→ 4 <i>bytes</i> , número de sequência da fonte
	→ 4 <i>bytes</i> , <i>time stamp</i> usado para computar latência na descoberta

Figura 9: Representação do pacote RREQ do AODV

dos nós intermediários (“C”, “D” e “E”) guarda como próximo salto na entrada da tabela o endereço do nó do qual receberam o primeiro RREQ. Essas entradas de rota reversa são armazenadas tempo suficiente para o RREQ chegar ao destino e produzir o RREP. A rota reversa é representada pelas setas tracejadas na Figura 8.

A rota direta é construída conforme o RREP é transmitido usando a rota reversa, até chegar ao nó fonte. Desde o nó destino ou um nó intermediário, cada nó ao longo do caminho reverso guarda na tabela de roteamento uma entrada para a rota direta ($A \rightarrow B$) apontando como próximo salto o nó do qual recebeu o RREP. Cada entrada de rota possui também um tempo de expiração e o número de sequência do destino. A rota direta é então dada pela rota reversa, que foi configurada usando os pacotes RREQ. No exemplo da Figura 8 a rota direta é: $A \rightarrow C \rightarrow D \rightarrow E \rightarrow B$.

Os nós que não recebem o RREP terão suas entradas da rota reversa expiradas e serão apagadas. Uma representação do RREP é mostrada na Figura 10.

	→ 1 <i>byte</i> , tipo do pacote
	→ 2 <i>bytes</i> , reservado
	→ 1 <i>byte</i> , contagem de saltos
	→ 4 <i>bytes</i> , endereço IP do destino
	→ 4 <i>bytes</i> , número de sequência do destino
	→ 4 <i>bytes</i> , endereço IP da fonte
	→ 4 <i>bytes</i> , tempo de vida do pacote
	→ 4 <i>bytes</i> , <i>time stamp</i> usado para computar latência na descoberta

Figura 10: Representação do pacote RREP do AODV

Cada nó possui uma tabela de roteamento para cada uma das rotas ativas que passe pelo mesmo. Tanto para a rota direta como para a rota reversa a estrutura da tabela é a mesma:

- endereço IP do destino;
- próximo salto;
- quantidade de saltos;
- número de sequência do destino;
- nós vizinhos ativos para a rota em questão;
- tempo de expiração para a entrada.

Um vizinho é considerado ativo se esse origina ou recebe um pacote de dados dentro do tempo de expiração de rota.

Se houver mais de uma entrada para o mesmo destino a de maior número de sequência será escolhida. Caso o número de sequência seja o mesmo a rota de menor quantidade de saltos é usada.

4.2 Manutenção de Rota

A manutenção de uma rota é composta de duas etapas: monitoração e correção. Na monitoração o correto funcionamento das rotas é constantemente verificado, possibilitando a detecção de perda de rota e a imediata tomada por medidas de correção. A monitoração de uma rota é realizada uma vez que essa seja construída e esteja funcionando, ou seja, estão finalizadas as etapas de obtenção das rotas direta e reversa.

A detecção de perda de um nó ativo, e conseqüente perda de rota, pode ser feita por meio dos chamados pacotes HELLO. Esses pacotes são enviados periodicamente em *broadcast* local pelos nós ativos. Seu intuito é sinalizar aos nós vizinhos o correto funcionamento do enlace.

Uma outra forma de detectar a perda de um nó ativo é por meio do pacote de ACK. Consecutivas perdas do pacote de ACK podem levar ao entendimento de quebra de enlace.

Quando um nó de uma rota ativa é perdido, uma mensagem de erro (ERROR) é gerada e enviada para notificar aos outros nós a perda de rota. A recepção do pacote ERROR pelo nó fonte leva ao reinício do ciclo de descoberta de rota.

4.3 Conclusão

O AODV permite o roteamento dinâmico entre nós móveis que compõem uma rede *ad hoc*. O AODV possibilita a cada nó obter rotas para novos destinos e não requer que os nós mantenham rotas para destinos que não estejam participando de nenhuma comunicação.

O AODV garante a ausência de *loops* de roteamento através da utilização de números de seqüência, evitando assim o problema da “contagem para infinito” característico dos protocolos que utilizam vetores de distância. Os números de seqüência também determinam o quão recente é uma rota.

Também vale ressaltar que as tabelas de roteamento de cada nó só possuem uma entrada para cada destino. Ao requisitar uma nova rota, automaticamente duas rotas são encontradas: rota direta e a rota indireta. A rota direta aponta para o nó de destino requisitado no pacote RREQ, enquanto que a rota indireta aponta para o nó fonte do pacote RREQ, que está requisitando a nova rota.

A métrica para seleção de rotas do AODV é o tempo de chegada do pacote RREQ. O primeiro pacote RREQ, identificado de maneira única pelo par (endereço no fonte; BID), que chega ao nó é adotado como rota reversa. A escolha da rota reversa corresponde na escolha da rota direta.

5 *Congestion-Aware AODV* (*CA-AODV*)

Neste capítulo são dados detalhes da implementação e funcionamento do CA-AODV. O CA-AODV foi projetado usando como base o AODV, discutido no Capítulo 4. O CA-AODV se diferencia do AODV apenas no processo de descoberta de rotas, seu mecanismo de manutenção das rotas é o mesmo do AODV.

Como pré-requisitos de projeto foi estabelecido que as modificações feitas deveriam acrescer o mínimo possível de carga à estrutura do AODV, sem perda de sua estabilidade e dinâmica. Então, o aumento no fluxo de pacotes de controle (RREQ, RREP, HELLO, ERROR) deve ser mínimo. Também não há acréscimo nesse conjunto de pacotes para sua operação, o CA-AODV usa os mesmos de pacotes de controle do AODV. Quanto às modificações nas estruturas desses pacotes, também foi buscado o mínimo acréscimo de campos.

5.1 Descoberta de Rotas

Como explicado no Capítulo 4, no AODV a análise dos pacotes RREQ, identificados pelo par (endereço do nó fonte, BID), é feita apenas para os pacotes que ainda não tenham sido recebidos, os RREQ que chegam “atrasados” são descartados. Para a rede vista na Figura 11 a rota de “A” a “B” é $A \rightarrow J \rightarrow G \rightarrow E \rightarrow B$. O nó “G” descarta o RREQ transmitido via rota $C \rightarrow D$ e a rota reversa adotada até o nó “A” é $J \rightarrow A$.

No CA-AODV não há o descarte imediato desses RREQ “atrasados”, esses são recebidos e processados. Para a Figura 11, a rota reversa para o RREQ “atrasado” ($D \rightarrow C \rightarrow A$) é então comparada, segundo a métrica de qualidade, com a rota adotada a priori ($J \rightarrow A$). Ao modificar a rota reversa durante o ciclo de busca a rota direta também é alterada. Para que a comparação possa surtir efeito é necessário que os RREQ “atrasados” sejam analisados apenas durante o ciclo de busca, ou seja, caso o RREP já tenha sido transmitido

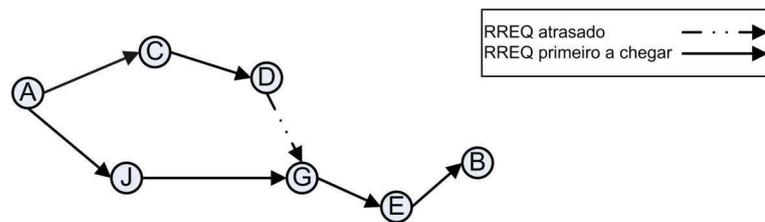


Figura 11: Descoberta de rotas

em nada adianta processar esses RREQ.

Quanto à qualidade das rotas, é alegado que o tempo de chegada não é suficiente para validar a superioridade de qualidade de uma rota. Ou seja, o fato de um RREQ ter chegado primeiro não significa necessariamente ter vindo por uma rota melhor, isso acontece caso as circunstâncias de favorecimento para o tempo chegada do RREQ sejam apenas transitórias. Dessa forma é buscado avaliar e comparar as rotas de uma forma mais profunda levando em conta seu histórico de funcionamento.

A estrutura do pacote RREQ para o CA-AODV é visualizada na Figura 12. Os campos em destaque (métrica de atraso de ACK acumulada, taxa de uso acumulada e quantidade de nós em uso) são destinados à composição da métrica de qualidade da rota. Esses campos são apenas preenchidos pelos nós já em uso e que portanto possuem medida de seu atraso do pacote de ACK e taxa de uso.

	→ 1 <i>byte</i> , tipo do pacote
	→ 2 <i>bytes</i> , reservado
	→ 1 <i>byte</i> , contagem de saltos
	→ 4 <i>bytes</i> , identificador do RREQ (BID)
	→ 4 <i>bytes</i> , endereço IP do destino
	→ 4 <i>bytes</i> , número de sequência do destino
	→ 4 <i>bytes</i> , endereço IP da fonte
	→ 4 <i>bytes</i> , número de sequência da fonte
	→ 4 <i>bytes</i> , <i>time stamp</i> usado para computar latência na descoberta
	→ 2 <i>bytes</i>, métrica de atraso de ACK acumulada ao longo da rota
	→ 2 <i>bytes</i>, taxa de uso acumulada dos nós na rota
	→ 1 <i>byte</i>, quantidade de nós em uso na rota

Figura 12: Representação do pacote RREQ do CA-AODV

O campo métrica de atraso de ACK acumulada do pacote RREQ é o somatório da métrica de cada um dos nós em uso ao longo da rota. A métrica de atraso de ACK individual dos nós é calculada como a média de um conjunto de amostras do atraso de ACK. Assim como para a métrica do atraso de ACK, a taxa de uso acumulada é computada e acrescentada ao pacote RREQ pelos nós que já estão em uso. Ao contribuir

com a métrica acumulada da rota o nó também atualiza o campo da quantidade de nós em uso.

A informação de qualidade de uma dada rota é propagada conforme o pacote RREQ é recebido e re-encaminhado com os campos destinados à métrica de qualidade atualizados com as medidas de cada nó intermediário. Por exemplo, na rede da Figura 11, o nó “C” ao receber o RREQ vindo de “A” atualiza os campos reservados à métrica e retransmite o RREQ atualizado.

Para o CA-AODV o processo de descoberta de rotas tem início da mesma forma do AODV, o nó fonte envia pacotes RREQ endereçados ao nó destino. Cada nó intermediário ao receber um novo RREQ procede da seguinte forma:

- Checa se possui informação de rota até o destino:
 - se possuir envia RREP de volta até o nó fonte;
 - caso não possua:
 - * guarda informação da rota reversa, até o nó fonte:
 1. endereço IP do nó destino (nó fonte do RREQ);
 2. endereço IP do último salto;
 3. identificador do pacote RREQ (BID);
 4. tempo de expiração da rota reversa;
 5. número de sequência do nó fonte;
 6. atraso acumulado da rota;
 7. taxa de uso acumulada da rota;
 8. quantidade de nós em uso na rota.
 - * caso o nó já esteja em uso por alguma rota, possui medida de sua disponibilidade, esse nó atualiza os campos do pacote RREQ relacionados com a métrica, métrica acumulada (atraso de ACK e taxa de uso) e quantidade de nós em uso na rota, e;
 - * re-envia o pacote RREQ em *broadcast*.

Ao receber um RREQ “atrasado”, a nova rota reversa é comparada com a rota existente na tabela de roteamento, caso a rota do pacote “atrasado” seja avaliada como melhor a tabela de roteamento é atualizada para a nova rota reversa. Apenas os RREQ “atrasados” que são avaliados como melhores que a rota reversa na tabela de roteamento

são retransmitidos. Em relação ao AODV há um aumento no fluxo de pacotes RREQ, esse aumento é proporcional a quantidade de pacotes RREQ “atrasados” aceitos.

Como discutido no Capítulo 3 adotou-se a disponibilidade de recursos como parâmetro para seleção de rotas, sejam esses recursos de *hardware* ou de uso do meio. A classificação das rotas tomada sob o ponto de vista da disponibilidade dos nós levou à adoção do atraso do pacote de ACK e da taxa de uso dos nós na rota como parâmetros de análise.

O atraso de ACK é tomado para pacotes de dados enviados em *unicast*, ou seja, quando os nós já estão em uso por alguma rota e portanto é possível avaliar sua disponibilidade de recursos por meio do pacote de ACK. A taxa de uso de maneira semelhante ao atraso de ACK é tomada para os nós já em uso, esse parâmetro dá ideia do quão utilizado está sendo o nó pelas suas respectivas rotas ativas.

Adya em [35] aplica um esquema semelhante para a classificação dos enlaces usando o tempo de resposta de um pacote prova. Para o CA-AODV também seria possível usar esse método e empregar um pacote de prova para verificar a métrica do atraso de ACK de um dado nó, assim não somente os nós já em uso por alguma rota teriam uma medida própria de disponibilidade. No entanto, tal prática se mostraria inviável devido o aumento de carga na rede, sobretudo porque a medida do atraso de ACK não pode ser tomada sobre um único pacote, mas sobre um conjunto de amostras como foi mostrado na Seção 3.4.

Outro fato que inviabiliza o uso dos pacotes de prova, é que devido seu tamanho o comportamento desses pode não refletir a resposta da rede para os pacotes de dados, levando a conclusões errôneas sobre o possível desempenho de uma rota quando essa fosse usada para a transmissão de pacotes de dados.

Ao receber um novo RREQ os nós em uso necessitam computar dados recentes de suas métricas de disponibilidade, atraso de ACK e taxa de uso.

Para o atraso de ACK, o resultado é obtido pela média de um conjunto de amostras do atraso. Para montar o conjunto de amostras cada nó possui uma estrutura do tipo fila (*first-in, first-out*) na qual guarda informações sobre os pacotes de dados enviados. As informações referentes aos pacotes são: identificador do pacote e instante de envio. Ao receber o pacote de ACK o nó verifica o instante de recebimento e calcula o atraso de ACK para o pacote. Há também um *timer* associado à pilha de pacotes que apaga os pacotes que não tenham recebido confirmação de recebimento. No padrão IEEE 802.15.4 ambas primitivas usadas no envio de dados *unicast*, MCPS-DATA.request e MCPS-DATA.confirm,

possuem um campo destinado à discriminação do pacote de dados ao qual as primitivas estão associadas.

A taxa de uso é atualizada a cada pacote de dados que o nó recebe. Ao receber um pacote, não importa sua fonte ou destino, seu instante de recebimento é computado e esse instante é guardado numa estrutura do tipo fila, nessa estrutura estão guardadas as informações dos últimos dez pacotes de dados recebidos. Uma vez requisitada a taxa de uso, ao receber um pacote RREQ, o nó verifica o intervalo de tempo de recebimento dos últimos dez pacotes e desse intervalo é obtida a sua taxa de uso.

Para o CA-AODV, assim como para o AODV, a rota direta é dada pela configuração da rota reversa. Por isso, a análise de qualidade das rotas deve ser feita na etapa de configuração da rota reversa. A rota reversa é construída conforme o RREQ é propagado até chegar ao destino, que gera o RREP.

Voltando ao exemplo dado no Capítulo 3, e ilustrado na Figura 13, toma-se a hipótese de que uma terceira rota de “L” a “O” seja necessária, neste caso duas rotas hipotéticas seriam $L \rightarrow C \rightarrow G \rightarrow O$ e $L \rightarrow K \rightarrow G \rightarrow O$.

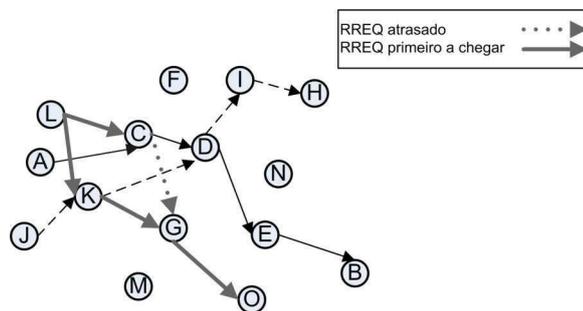


Figura 13: Rede com duas rotas

Para o AODV a rota encontrada seria $L \rightarrow K \rightarrow G \rightarrow O$, pois o nó “G” descarta o RREQ vindo por “C”, por haver chegado depois. No CA-AODV a escolha feita por “G” depende dos parâmetros de qualidade, contidos nos respectivos pacotes RREQ, de cada uma das rotas. Para que o RREQ “atrasado” possa ser analisado, esse deve ser recebido e processado por “G” antes do RREP, gerado pelo destino (nó “O”), passar por “G”.

Caso “G” entenda que a rota vinda por “C” seja melhor que a rota por “K” a rota reversa com destino a “L” será reconfigurada apontando para “C”, essa atualização da rota reversa também modifica a rota direta.

Não há muita diferença entre a configuração da rota direta do CA-AODV e do AODV. Usando a rota reversa, o RREP é transmitido do nó destino ou de um nó intermediário

até o nó fonte, nó “L” para a Figura 13, e cada nó ao longo da rota reversa configura o nó do qual recebeu o RREP como próximo salto na sua rota direta, que leva ao nó “O” na Figura 13.

A única diferença é que ao pacote RREP foi adicionado um campo de identificação do RREQ que originou o RREP. Esse campo, preenchido com BID do RREQ, é utilizado como uma medida contra *loops*, como será explicado mais a frente. A estrutura do RREP é apresentada na Figura 14.

	→ 1 <i>byte</i> , tipo do pacote
	→ 2 <i>bytes</i> , reservado
	→ 1 <i>byte</i> , contagem de saltos
	→ 4 <i>bytes</i> , endereço IP do destino
	→ 4 <i>bytes</i> , número de sequência do destino
	→ 4 <i>bytes</i> , endereço IP da fonte
	→ 4 <i>bytes</i> , tempo de vida do pacote
	→ 4 <i>bytes</i> , <i>time stamp</i> usado para computar latência na descoberta
	→ 4 <i>bytes</i>, identificador do RREQ que originou o RREP

Figura 14: Representação do pacote RREP do CA-AODV

O RREP assim como o RREQ é identificado de maneira única pelos campos endereço IP do destino (nó fonte do RREQ) e identificador do RREQ que originou o RREP (BID do RREQ).

A tabela de roteamento do CA-AODV é similar à do AODV, como é visualizado na Figura 15. Os campos em destaque são usados para análise de RREQ “atrasados” e para o mecanismo de eliminação de *loops*.

	→ 4 <i>bytes</i> , destino da rota
	→ 4 <i>bytes</i> , número de sequência
	→ 1 <i>byte</i> , contagem de saltos
	→ 4 <i>bytes</i> , endereço IP do próximo salto
	→ 4 <i>bytes</i> , tempo de expiração da entrada
	→ 2 <i>bytes</i>, métrica de atraso acumulada da rota
	→ 2 <i>bytes</i>, taxa de uso acumulada dos nós na rota
	→ 1 <i>byte</i>, quantidade de nós em uso na rota
	→ 4 <i>bytes</i> , próximo salto do primeiro RREQ recebido
	→ 1 <i>bit</i>, flag de sinalização de RREP já enviado

Figura 15: Tabela de roteamento do CA-AODV

Como já foi mencionado a análise de um RREQ “atrasado” só poderá surtir efeito na construção da rota se sua recepção e análise forem feitas antes da passagem do RREP. O

campo “*flag* de sinalização de RREP já enviado” tem a função de regular essa análise de RREQ “atrasados”.

Ao configurar uma nova entrada na tabela de roteamento ou ao atualizar para uma nova rota, os parâmetros de qualidade da rota são armazenados nos respectivos campos da tabela de roteamento (as métricas acumuladas de disponibilidade da rota e a quantidade de nós em uso).

Na tabela de roteamento, o campo “endereço IP do próximo salto” define a rota a seguir até o correspondente destino. Ao receber um RREQ “atrasado” a rota configurada a priori pelo primeiro RREQ recebido pode vir a ser substituída. Contudo, em caso de substituição de rota esse caminho adotado a priori não é descartado, é guardado no campo “próximo salto do primeiro RREQ recebido”, pois pode ser necessário para o mecanismo de eliminação de *loops*.

5.2 Mecanismos: Prevenção Contra *Loops* e Detecção e Eliminação de *Loops*

A não formação de *loops* é uma característica bastante importante do AODV. Um dos requisitos para um protocolo de roteamento é que esse seja livre de *loops*.

O CA-AODV modificou um comportamento característico do AODV que é o descarte de pacotes RREQ “atrasados”. A possibilidade de encontrar rotas melhores vem com o custo da possível formação de *loops* na rota reversa. Observando o cenário exemplo da Figura 16, verifica-se que para o CA-AODV caso “C” aceite o RREQ “atrasado” vindo de “D” e “D” aceite o RREQ vindo de “E” haverá a formação de *loop* para a rota reversa dado por $E \rightarrow C \rightarrow D \rightarrow E$.

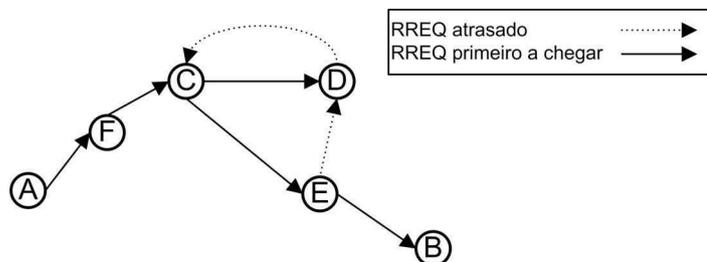


Figura 16: Formação de *loop*

Há duas formas para resolver esse problema:

- prevenir a formação do *loop*;
- detectar a formação e corrigi-la.

A priori, prevenir traz mais benefícios pois elimina o problema na sua origem. Contudo, o sistema de prevenção não deve inserir carga de maneira extensiva na rede nem limitar em demasia o sistema original.

A prevenção contra *loops*, pode ser conseguida limitando a análise dos pacotes RREQ “atrasados” àqueles cuja quantidade de saltos seja menor ou igual à quantidade de saltos do primeiro RREQ recebido, com isso é impedido o fechamento do *loop*. Esse método de prevenção vem com o custo de restrição no funcionamento do protocolo, pois a quantidade de rotas alternativas passíveis de análise é diminuída.

A detecção e correção é uma alternativa interessante, somente se essa tarefa não demandar um esforço por parte dos nós que signifique em aumento de carga na rede nem a formação do *loop* provoque, durante sua existência, grande depreciação no desempenho da rede.

Uma forma de detectar a formação do *loop* é por intermédio do pacote RREP. O pacote RREP gerado por “B” usará a seguinte rota reversa $B \rightarrow E \rightarrow D \rightarrow C \rightarrow E$. O nó “E”, por meio do RREP, é o primeiro nó a detectar a formação do *loop* e tomar as medidas de correção. O pacote RREP é identificado de forma única por meio de seus campos “endereço IP do destino” e “identificador do RREQ que originou o RREP”, assim, verificada a passagem de um RREP repetido estará caracterizada a formação do *loop*.

Uma vez detectado o *loop* é necessário corrigi-lo. Como medida de remediação a formação foi acrescentado o campo “próximo salto do primeiro RREQ recebido” na tabela de roteamento (Figura 15), que é a rota reversa configurada caso o mecanismo de rejeição de RREQ do AODV estivesse sendo usado. Como já foi dito o AODV tem como característica a não formação de *loop* portanto adotá-lo, nesse caso, significa fazer uso dessa sua propriedade.

Para a rede da Figura 16 quando “E” detecta o laço *loop* toma “C” como próximo salto, que ao receber o RREP também detecta o laço e toma “F” como próximo salto que faz com que o RREP chegue até “A”. Como resultado a rota reversa selecionada entre “B” e “A” é $B \rightarrow E \rightarrow C \rightarrow F \rightarrow A$ e a direta é $A \rightarrow F \rightarrow C \rightarrow E \rightarrow B$.

A rota adotada pelo mecanismo do CA-AODV só é confirmada com a chegada do RREP a fonte, nesse instante é garantida a não formação de *loops* tanto na rota direta

como na reversa.

Usar o mecanismo de rejeição do pacote RREQ do AODV não significa necessariamente que obtém-se a mesma rota caso o AODV estivesse em uso. O pacote RREQ do CA-AODV é maior do que o RREQ do AODV, isso lhe confere um comportamento diferente na formação da rotas.

Os resultados a serem apresentados neste trabalho foram obtidos tanto usando o mecanismo de prevenção como o de detecção e eliminação de *loops*.

5.3 Mecanismo de Análise das Rotas

O mecanismo de análise define se um dado nó aceita ou não um pacote RREQ “atrasado”, baseado nas respectivas medidas de disponibilidade das rotas. Ao receber um RREQ “atrasado”, o nó faz uso dos campos “métrica de atraso de ACK acumulada”, “taxa de uso acumulada” e “quantidade de nós em uso” do pacote RREQ e da tabela de roteamento para comparar as rotas.

Foram definidas situações nas quais a análise de RREQ “atrasados” é inviável, sendo esses sumariamente descartados nos seguintes casos:

- expirado o tempo de recebimento de pacotes RREQ “atrasados”;
- o nó é o destino do RREQ;
- o RREP para o dado RREQ já foi enviado;
- o próximo salto na tabela de roteamento é o nó fonte.

Ao receber o primeiro RREQ cada nó configura o tempo de atraso limite, intervalo no qual os RREQ “atrasados” estão aptos à análise. Embora neste trabalho seja alegado que o tempo de chegada do RREQ não é o melhor parâmetro para análise das rotas, há de certo uma dependência desse parâmetro com a qualidade da rota. Assim não é recomendável desconsiderar totalmente o tempo de chegada, por isso é adotado um atraso máximo no qual os RREQ “atrasados” são aceitos para análise.

O nó destino ao receber um RREQ gera o RREP e o envia com um atraso de 100 *ms*, esse atraso possibilita aos nós intermediários receber e analisar um número maior de rotas alternativas. Para evitar a sobrecarga de pacotes RREP os nós de destino são

impossibilitados de analisar RREQ “atrasados”, os quais caso fossem aceitos ocasionariam o envio de outros RREP.

Discutido na Seção 5.1, o *flag* de envio do pacote RREP sinaliza a ineficácia da análise de RREQ “atrasados” na construção das rotas.

Caso o próximo salto na tabela de roteamento seja o próprio nó fonte, a análise de rotas alternativas até o nó fonte é inviável.

Os pacotes RREQ “atrasados” que não forem descartados serão então comparados com a rota existente na tabela de roteamento. A informação dos parâmetros de disponibilidade tanto do RREQ como da tabela de roteamento dizem respeito à rota reversa, como a rota direta corresponde a reversa esses parâmetros também estão relacionados à rota direta.

5.3.1 Parâmetros de Disponibilidade

Os parâmetros usados na composição da métrica de disponibilidade de uma dada rota, a taxa de uso acumulada dos nós e métrica acumulada do atraso, referem-se à disponibilidade de *hardware* e do meio de transmissão.

A taxa de uso está relacionada com *hardware*, ao comparar duas rotas a taxa de uso dos nós de cada uma das rotas dá ideia do quão escassos são seus recursos. Esse parâmetro leva em conta a intensidade do tráfego de dados individual de cada nó.

A métrica acumulada do atraso está tanto relacionada com a disponibilidade de *hardware* como a disponibilidade do meio, como já discutido no Capítulo 3. Quanto ao *hardware* o atraso de ACK é proporcional ao grau de preenchimento da pilha dos dispositivos, quanto ao meio o parâmetro reflete seu grau de congestionamento. Assim como para a taxa de uso acumulada dos nós, o grau de utilização do recurso, *hardware* e meio, tem influência na métrica de atraso acumulada.

5.3.2 Composição da Métrica de Disponibilidade

Ao receber um pacote RREQ “atrasado” o nó deve compor os parâmetros de disponibilidade numa única medida com a qual consiga classificar uma rota como melhor que outra.

O parâmetro métrica de atraso acumulada é proporcional à quantidade de nós em uso, quanto maior a quantidade de nós em uso mais medidas do atraso de ACK são tomadas, que se reflete no atraso acumulado. Sejam duas rotas “1” e “2” com suas respectivas

métricas de atraso acumulado ma_1 e ma_2 e com quantidade de nós em uso respectivamente q_1 e q_2 . A análise da métrica de atraso não deve ser feita de maneira absoluta. Ao comparar a rota “1” com a “2”, o fato de ter $ma_1 > ma_2$ não necessariamente exprime que a rota “1” tem em seu histórico de atraso de ACK acumulado uma maior dificuldade na transmissão, significando que “2” seja melhor que “1”. A intenção em analisar a métrica acumulada é identificar se em algum ponto, ou pontos, da rota existe um maior congestionamento que tenha impacto negativo no seu desempenho. A métrica de atraso de uma rota deve, portanto, ser analisada por nó, usando uma métrica de atraso médio. A métrica de atraso médio das rotas “1” (m_1) e “2” (m_2) será:

$$m_1 = \frac{ma_1}{q_1}$$

$$m_2 = \frac{ma_2}{q_2}$$

A taxa de uso acumulada dos nós para as rotas 1 e 2, respectivamente ur_1 e ur_2 , diferentemente do atraso de ACK é analisada de maneira absoluta, ou seja, não é tomada uma medida média. Para facilitar o tratamento dos parâmetros, esses serão avaliados de forma percentual, Equações 5.1, 5.2, 5.3 e 5.4.

$$m_{1\%} = \frac{m_1}{m_1 + m_2} \quad (5.1)$$

$$m_{2\%} = 1 - m_{1\%} \quad (5.2)$$

$$ur_{1\%} = \frac{ur_1}{ur_1 + ur_2} \quad (5.3)$$

$$ur_{2\%} = 1 - ur_{1\%} \quad (5.4)$$

A métrica final usada para classificar uma rota como melhor que outra é produto da composição dos resultados das Equações 5.1, 5.2, 5.3 e 5.4. Uma forma intuitiva de montar a métrica final é por uma composição ponderada dos parâmetros percentuais:

$$\text{(Métrica final da rota 1)} \quad mf_1 = P_m \times m_{1\%} + P_{ur} \times ur_{1\%} \quad (5.5)$$

$$\text{(Métrica final da rota 2)} \quad mf_2 = P_m \times m_{2\%} + P_{ur} \times ur_{2\%} \quad (5.6)$$

onde,

P_m é o fator de ponderação da métrica de atraso médio;

P_{ur} é o fator de ponderação da taxa de uso acumulada na rota.

Se $mf_1 < mf_2$ conclui-se que a rota “1” possui maior disponibilidade de recursos que a rota “2”, caracterizando “1” como a melhor escolha. Quanto maior o fator de ponderação maior relevância terá o correspondente parâmetro de qualidade na composição da métrica final.

Na análise de disponibilidade das rotas busca-se dois objetivos: minimizar métrica média do atraso de ACK e minimizar a taxa de uso acumulada dos nós na rota. O ajuste dos fatores de ponderação define o quão relevante é cada um desses objetivos na tomada de decisão.

5.3.3 Composição da Métrica Usando Modelo FUZZY

Dadas as incertezas na avaliação dos parâmetros recebidos, aplicou-se um modelo *fuzzy* [36] de análise dos parâmetros na composição da métrica final. A lógica *fuzzy* trata de problemas nos quais é imprecisa a avaliação dos parâmetros, como é o caso do CA-AODV na comparação da qualidade de duas rotas baseada nos dados de disponibilidade de cada uma dessas. A questão é como tratar os parâmetros obtidos pelas equações 5.1, 5.2, 5.3 e 5.4, para efetuar a tomada de decisão.

A análise de cada um dos parâmetros tem um objetivo próprio. As equações 5.3 e 5.4 estão relacionadas com a escolha de rotas cujos nós estejam sendo menos utilizados. As equações 5.1 e 5.2 estão também relacionadas à seleção de rotas cujos nós estejam menos utilizados pelas rotas existentes, mas também tangem fatores relacionados ao ambiente no qual os nós se encontram.

Um interessante modelo para análise de dados nebulosos é apresentado por Ross em [37]. Uma das técnicas apresentadas visa a tomada de decisões com múltiplos objetivos, baseada na análise de dados nebulosos.

No CA-AODV, cada nó ao receber um RREQ “atrasado” deve comparar e decidir qual rota tomar, a rota da tabela de roteamento ou a rota do RREQ “atrasado”. A avaliação dessas objetiva encontrar rotas com uma maior disponibilidade de recursos, e que portanto proporcionem melhores taxas de desempenho. Para chegar a esse objetivo são analisados dois parâmetros, a taxa de uso e o atraso médio de ACK. Cada um dos parâmetros analisa uma diferente característica da rota, e a escolha deve ser feita buscando rotas que alcancem da melhor forma possível cada um dos seguintes objetivos:

- minimizar o atraso de ACK médio da rota;
- minimizar a taxa de uso dos nós na rota.

Há dois problemas primários na resolução de problemas com múltiplos objetivos, o primeiro está na aquisição de dados que considerem a satisfação dos diferentes objetivos e o segundo está na ponderação de cada um dos objetivos na decisão final [37].

Um problema típico envolve a seleção de uma alternativa, a_i , dentro de um universo de alternativas “ A ”, $\{a_1, a_2, \dots, a_j\}$, dado um conjunto “ O ”, $\{o_1, o_2, \dots, o_i\}$, de critérios ou objetivos importantes na tomada de decisão. É desejado avaliar o quanto cada alternativa satisfaz cada objetivo e combinar a ponderação dos objetivos numa única função de decisão.

Para o CA-AODV temos as seguintes definições para os conjuntos “ A ” e “ O ”, respectivamente alternativas e objetivos:

$$A = \{a_1, a_2\} = \{\text{rota da tabela de roteamento, nova rota do RREQ “atrasado”}\}$$

$$O = \{o_1, o_2\} = \{\text{minimizar o atraso de ACK médio, minimizar a taxa de uso dos nós na rota}\}$$

O próximo passo é determinar a função de pertinência de cada uma das alternativas em cada objetivo, representado por $\mu_{o_n}(a_m)$. A função de pertinência retorna o quanto cada uma das alternativa está satisfazendo cada um dos objetivos.

No CA-AODV as variáveis *fuzzy* são obtidas das equações 5.1, 5.2, 5.3 e 5.4. A determinação do valor de pertinência de um parâmetro num dado conjunto, expressa o quanto esse parâmetro está satisfazendo os pré-requisitos do dado conjunto.

Assim, toma-se por exemplo uma rota “1” (a_1), com seus respectivos parâmetros de análise: $m_{1\%}$ e $ur_{1\%}$. O parâmetro $m_{1\%}$ é usado para analisar a rota quanto ao primeiro objetivo, minimizar o atraso de ACK médio, e $ur_{1\%}$ analisa quanto ao segundo objetivo, minimizar a taxa de uso dos nós na rota. Ambos os parâmetros são inversamente proporcionais aos respectivos objetivos, ou seja, quanto maior $m_{1\%}$ menos satisfeito é o objetivo de minimizar o atraso de ACK médio. Como $m_{1\%}$ e $ur_{1\%}$ estão no intervalo $[0; 1]$, suas funções de pertinência são apresentadas nas Equações 5.7 e 5.8.

$$\mu_{o1}(a_1) = 1 - m_{1\%} \quad (5.7)$$

$$\mu_{o2}(a_1) = 1 - ur_{1\%} \quad (5.8)$$

Quanto menor for seu parâmetro percentual, mais satisfeita estará a condição para que a alternativa alcance o respectivo objetivo.

Uma vez determinadas as funções de pertinência dos parâmetros nos seus respectivos objetivos, o último passo é compor a métrica final para tomada de decisão. A partir dos parâmetros percentuais de cada rota é verificado o quanto cada uma satisfaz cada objetivo, o próximo passo é definir o quão importante é cada objetivo na tomada de decisão final. A decisão é tomada usando as Equações 5.9 e 5.10.

$$\text{(Métrica final da rota 1)} \quad mf_1 = \min\{\max[(1 - P_m), \mu_{o1}(a_1)], \max[(1 - P_{ur}), \mu_{o2}(a_1)]\} \quad (5.9)$$

$$\text{(Métrica final da rota 2)} \quad mf_2 = \min\{\max[(1 - P_m), \mu_{o1}(a_2)], \max[(1 - P_{ur}), \mu_{o2}(a_2)]\} \quad (5.10)$$

onde,

P_m é o fator de ponderação da métrica de atraso médio;

P_{ur} é o fator de ponderação da taxa de uso acumulada na rota.

Para a tomada de decisão o maior resultado da expressão define a melhor opção, ou seja, se $mf_1 > mf_2$ então a opção “1” é escolhida.

Yager em [38], explica melhor as Expressões 5.9 e 5.10. Para um determinado objetivo, a negação de sua importância ($1 - P_m$ e $1 - P_{ur}$) atua como uma barreira, tal que todas as alternativas abaixo dessa barreira se tornam iguais ao valor dessa barreira. Os valores abaixo da barreira são desconsiderados, apenas os valores acima dessa barreira têm relevância no valor final da expressão.

Os resultados a serem apresentados no Capítulo 6 são para o uso do mecanismo de decisão empregando o modelo *fuzzy*.

5.4 Conclusão

O CA-AODV é uma implementação bastante próxima do AODV, diferenciando-se basicamente pelo mecanismo que analisa os pacotes RREQ. O CA-AODV trabalha assim como o AODV sobre a rota obtida pelo primeiro RREQ a chegar, a diferença é a possibilidade de adoção de uma rota alternativa caso essa que seja considerada melhor por ter maior disponibilidade de recursos. A rota alternativa vem da análise dos RREQ “atrasados”.

A análise de RREQ “atrasados” possibilita a formação de *loops*. Foram adotados dois procedimentos para eliminação de *loops*: 1) preventivo; e 2) corretivo. Cada um desses mecanismos possui suas vantagens e desvantagens. O mecanismo preventivo possui uma maior eficiência na eliminação de *loops*, em contrapartida o universo de análise de rotas alternativas é restringido. O mecanismo corretivo tem o contratempo de ter que detectar e eliminar os possíveis *loops*, contudo sua capacidade de encontrar rotas alternativas é maior que no mecanismo preventivo. Com isso pode-se dizer que o modelo do CA-AODV não possui *loops* ao final da busca pela rota.

Para que a adoção da rota alternativa seja efetiva na construção da rota final, é preciso que o pacote de RREP chegue até algum nó que tenha optado pela a rota do RREQ “atrasado” para sua rota reversa.

O CA-AODV trabalha apenas na fase de construção da rota, a análise da rota via a métrica de disponibilidade não é usada para supervisão de seu desempenho. Isto significa que uma vez construída uma rota, o CA-AODV não age em caso de depreciação do desempenho da rota, haverá medidas de busca de um nova rota apenas no caso de quebra do enlace.

O mecanismo de análise das rotas processa os parâmetros de disponibilidade vindos no pacote RREQ: 1) atraso de ACK acumulado; 2) taxa de uso acumulada; e 3) quantidade de nós em uso. Esses três parâmetros devem ser compostos numa única métrica para a classificação de qualidade da rota.

Ao analisar a métrica de atraso de ACK acumulada, deseja-se identificar se em algum nó da rota existe um maior congestionamento que tenha impacto negativo no seu desempenho. Assim esse parâmetro é tratado na média por salto. A taxa de uso acumulada diferentemente do atraso de ACK é analisada de maneira absoluta.

O mecanismo de seleção das rotas consiste em escolher entre duas rotas, a antiga na

tabela de roteamento e a nova do pacote RREQ, a que melhor alcança de forma ponderada dois objetivos: 1) encontrar rotas que minimizem o atraso de ACK médio; e 2) encontrar rotas que minimizem a taxa de uso dos nós na rota.

6 *Simulações: Resultados e Discussões*

Neste capítulo são apresentados e discutidos os resultados de exaustivos testes por simulação do CA-AODV, verificando em diferentes cenários a eficiência do CA-AODV quando comparado ao AODV. Foi empregado como ferramenta de simulação o *Network Simulator* (NS-2 [12] [13], versão 2.32). O pós processamento dos resultados do NS-2 é feito por ferramentas construídas em *script bash*.

Para a comparação de desempenho entre o CA-AODV e AODV foram adotados os seguintes parâmetros de análise:

- Razão de entrega de pacotes (*Packet Delivery Ratio* - PDR): razão entre a quantidade de pacotes de dados devidamente recebidos no destino e a quantidade total de pacotes de dados enviados pelo nó fonte;
- Atraso médio para a chegada dos pacotes de dados;
- Razão de perda de pacotes: razão entre a quantidade de pacotes de dados perdidos e o total de envios de pacotes de dados.

Esses parâmetros de análise referem-se aos pacotes de dados, ou seja, são computados somente durante o funcionamento das rotas, não contabilizando o desempenho dos protocolos na sua construção.

A política adotada para comparação de desempenho é confrontar os resultados do CA-AODV com e sem habilitação do mecanismo de análise dos RREQ “atrasados”, para então avaliar seu desempenho ante o AODV. O chaveamento do mecanismo de análise dos RREQ no exame dos resultados reforça a avaliação da eficiência desse mecanismo, verificando se o parâmetro de análise das rotas (disponibilidade de recursos) e o mecanismo de análise são capazes de otimizar a escolha das rotas.

Ao desabilitar o mecanismo de análise dos RREQ “atrasados” o CA-AODV se torna

o AODV com pacotes de controle maiores. Devido à diferença na carga dos pacotes de controle, as rotas encontradas pelo CA-AODV sem o mecanismo de análise são a priori diferentes das rotas do AODV. O que o mecanismo de análise faz é procurar otimizar essa rota inicialmente obtida. Caso nenhum RREQ “atrasado” seja aceito a rota adotada é a mesma para o mecanismo de análise desativado.

6.1 Resultados das Simulações

Os cenários 1 e 2 apresentados nesta seção, foram construídos de forma a obter e avaliar o desempenho em situações que ressaltam as principais características do CA-AODV:

- evitar rotas cujos nós e meio estejam congestionadas;
- diminuir a concentração das rotas num mesmo conjunto de nós, distribuindo melhor a carga na rede.

A escolha por rotas menos congestionadas e com menos interferência vem da minimização do termo referente à métrica de atraso médio de ACK da Equação 5.5. Ao dar maior ênfase ao fator de ponderação P_m , as menores diferenças entre os parâmetros da métrica de atraso serão mais relevantes. Rotas menos congestionadas resultam na diminuição da perda de pacotes que em consequência reduz a necessidade de retransmissão.

A desconcentração das rotas, é alcançada pela análise do parâmetro “taxa de uso acumulada dos nós”. Enfatizar esse parâmetro na Equação 5.5, propicia a distribuição das rotas nos nós na rede. Ao distribuir as rotas nos nós da rede também é distribuído o consumo de energia da rede, essa distribuição das rotas significa também a diminuição da perda de pacotes.

Esses dois objetivos têm impacto direto na vida útil da rede.

6.1.1 Cenário 1

Neste cenário foi destacado o desempenho do CA-AODV na construção de rotas que busquem ambientes menos congestionados. Na rede representada na Figura 17, há basicamente dois caminhos, $9 \leftrightarrow 10 \leftrightarrow 11$ ou $2 \leftrightarrow 3 \leftrightarrow 4$, ligando as duas extremidades da rede. Os nós “27” e “28” são destinados a gerar interferência nas proximidades do nó “3”.

A topologia adotada neste cenário é um modelo para o funcionamento de uma rede real, pode acontecer da topologia real ser diferente da apresentada na Figura 17 mas

seu comportamento se aproximar desse. Isso decorre de diferentes situações, como na existência de objetos que impeçam a livre comunicação entre os ramos da rede ($9 \leftrightarrow 10 \leftrightarrow 11$ e $2 \leftrightarrow 3 \leftrightarrow 4$), caso os dois ramos estejam em planos distintos, etc.

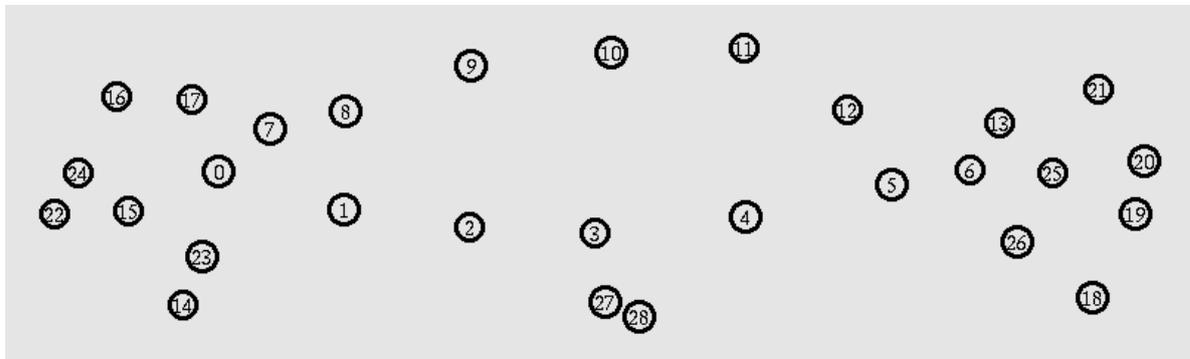


Figura 17: Primeiro cenário de testes

O objetivo deste teste é verificar se o mecanismo é capaz de evitar uma rota congestionada, passando pelo nó “3”, quando a rota naturalmente escolhida (mecanismo de análise desativado) usa o trecho de rede afetado pela interferência.

Como o ambiente de estudo se caracteriza por haver fontes de interferência no meio agindo de maneira não homogênea na rede foi dado maior peso à métrica de atraso médio na decisão, $P_m > P_{ur}$ ($0,7 > 0,4$). O mecanismo do CA-AODV deve então procurar rotas que fujam de nós nas proximidades da interferência. Por ser um experimento demonstrativo das características principais do CA-AODV, é usado um cenário simples com apenas três conexões iniciadas em sequência ($14 \rightarrow 21$, $17 \rightarrow 18$, $19 \rightarrow 16$) além da conexão de interferência entre $27 \rightarrow 28$, todas as conexões enviam dados a taxas constantes.

O experimento foi conduzido segundo a configuração apresentada na Tabela 2.

Tabela 2: Configuração do Cenário 1

conexão	carga útil (<i>bytes</i>)	taxa (kbps)	instante início (s)	instante fim (s)
$14 \rightarrow 21$	10	1,00	56	320
$17 \rightarrow 18$	10	0,60	72	320
$19 \rightarrow 16$	10	0,85	88	320
$27 \rightarrow 28$	10	7,80	80	320

Como a conexão de interferência, $27 \rightarrow 28$, é iniciada no instante 80 s seu efeito só aparecerá na tomada de decisão para formação da última conexão, $19 \rightarrow 16$.

Para a validação dos resultados de um experimento, esses devem atender ao princípio da repetibilidade na reprodução do experimento. Para tal, foi adotado o método de

análise estatística dos resultados apresentado no Capítulo 3, no qual um conjunto de diferentes experimentos independentes são conduzidos para um mesmo cenário. Os resultados encontrados são apresentados na Tabela 3.

Tabela 3: Resultados para o Cenário 1

CA-AODV				
modo de uso	PDR	razão de perdas	atraso médio	n_{min}
mecanismo de análise desabilitado	0,60185 (0,57176 - 0,63195)	0,23685 (0,22501 - 0,24869)	0,03998 (0,03799 - 0,04199)	58,168 \approx 60
mecanismo de análise habilitado + mecanismo de correção de <i>loops</i>	0,62040 (0,58938 - 0,65142)	0,22580 (0,21451 - 0,23709)	0,04011 (0,03811 - 0,04212)	60,574 \approx 61
mecanismo de análise habilitado + mecanismo de prevenção de <i>loops</i>	0,60981 (0,57932 - 0,64030)	0,23522 (0,22346 - 0,24698)	0,03985 (0,03786 - 0,04184)	61,720 \approx 62
AODV				
	0,60364 (0,57345 - 0,63382)	0,23476 (0,22302 - 0,24650)	0,03969 (0,03770 - 0,04167)	63,669 \approx 64

Para o CA-AODV usando o mecanismo corretivo contra *loops*, é verificado ganho de desempenho tanto quando confrontado com o CA-AODV sem o mecanismo de análise como quando confrontado com o AODV. Para ambos, o CA-AODV apresenta uma melhora de desempenho de $\approx 3,0\%$ no PDR e de $\approx 4,5\%$ na razão de perdas. Quanto ao desempenho do sistema quando usado os mecanismo de prevenção de *loops*, houve um ganho bastante pequeno ficando o desempenho do sistema nos níveis do desempenho do CA-AODV sem o mecanismo de análise e do AODV.

Para o atraso médio, alcançou-se apenas um desempenho similar em todos os casos.

6.1.2 Cenário 2

O primeiro cenário tinha o intuito de verificar a eficácia e o ganho do CA-AODV na seleção de rotas sob interferência. Este segundo cenário, mostrado na Figura 18, tem intuito de verificar o funcionamento do segundo parâmetro de análise das rotas, a taxa de uso acumulada dos nós. Assim, foi adotado 0,4 e 0,7 respectivamente para os fatores de ponderação P_m e P_{ur} .

Foi usado um cenário muito próximo do cenário 1, a exceção dos nós “27” e “28” usados para criar interferência.

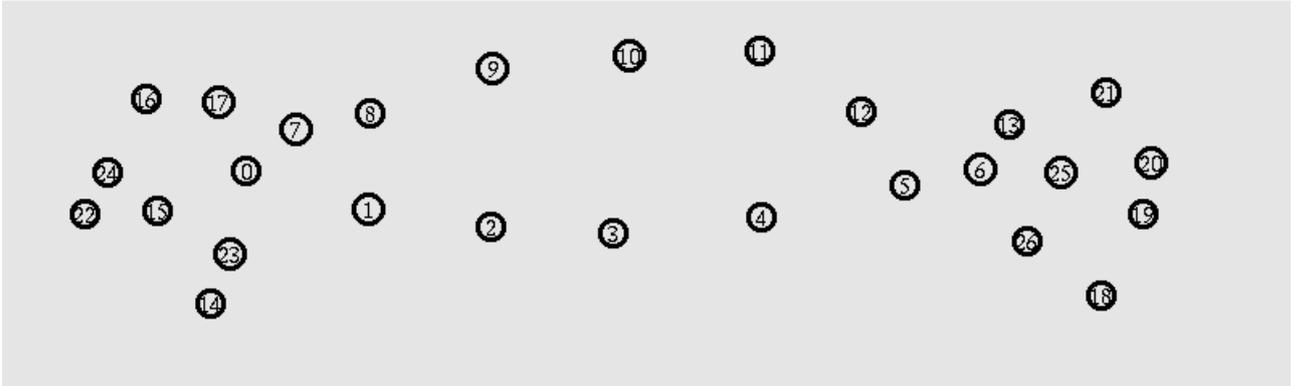


Figura 18: Segundo cenário de testes

O experimento foi conduzido segundo a configuração apresentada na Tabela 4.

Tabela 4: Configuração do Cenário 2

conexão	carga útil (<i>bytes</i>)	taxa (kbps)	instante início (s)	instante fim (s)
16 → 12	10	1,00	56	320
20 → 22	10	1,00	72	320

Os resultados encontrados são apresentados na Tabela 5.

Tabela 5: Resultados para o Cenário 2

CA-AODV				
modo de uso	PDR	razão de perdas	atraso médio	n_{min}
mecanismo de análise desabilitado	0,91724 (0,87138 - 0,96310)	0,06228 (0,05917 - 0,06539)	0,02947 (0,02799 - 0,03094)	25,195 ≈ 26
mecanismo de análise habilitado + mecanismo de correção de <i>loops</i>	0,97440 (0,92568 - 1,00)	0,03290 (0,03125 - 0,03454)	0,02825 (0,02684 - 0,02966)	8,715 ≈ 9
mecanismo de análise habilitado + mecanismo de prevenção de <i>loops</i>	0,92008 (0,87408 - 0,96609)	0,05976 (0,05677 - 0,06274)	0,02939 (0,02792 - 0,03086)	24,286 ≈ 25
AODV				
	0,93598 (0,88918 - 0,98278)	0,05245 (0,04982 - 0,05507)	0,02936 (0,02789 - 0,03083)	19,701 ≈ 20

O CA-AODV quando usado com o mecanismo corretivo contra *loops*, proveu ganho de desempenho em relação ao CA-AODV com o mecanismo de análise desativado e em relação ao AODV. Para o CA-AODV sem o mecanismo de análise, é verificado um ganho de ≈ 6,0% nos resultados do PDR e de ≈ 45,0% na razão de perdas. Quanto ao AODV o CA-AODV apresentou uma melhora de ≈ 4,1% nos resultados do PDR e de ≈ 36,0%

na razão de perdas. Quanto ao desempenho do sistema quando usando os mecanismo de prevenção de *loops*, é verificada a melhora no desempenho da rota selecionada quando comparado com o sistema sem o mecanismo de análise da rotas, no entanto, seu ganho não foi capaz de superar os resultado do AODV.

6.1.3 Cenário 3

Neste cenário, mostrado na Figura 19, é pretendido verificar o comportamento do CA-AODV em situações nas quais as rotas tendem a se concentrar num único conjunto de nós.

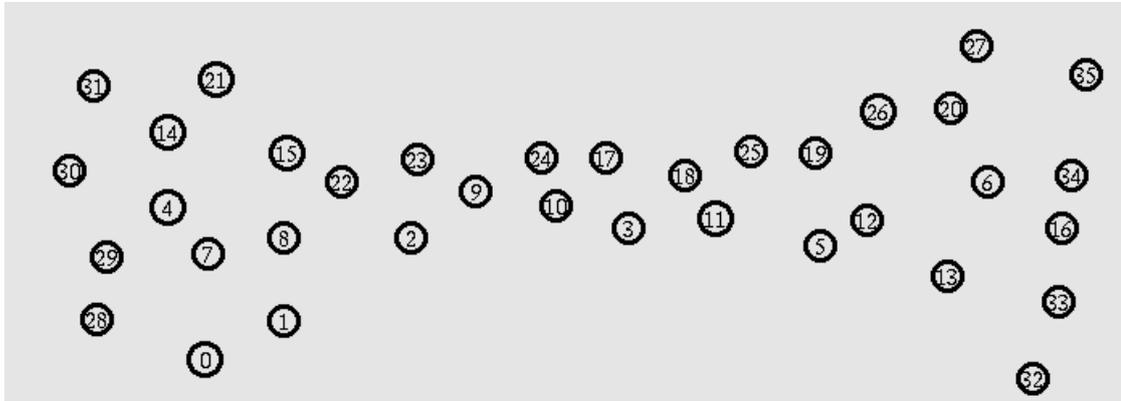


Figura 19: Terceiro cenário de testes

O experimento foi conduzido em duas etapas (Experimento1 e Experimento2) com diferentes taxas transmissão, segundo a configuração apresentada na Tabela 6.

Tabela 6: Configuração do Cenário 3

conexão	carga útil (<i>bytes</i>)	instante início (s)	instante fim (s)	taxa (kbps)	
				Experimento1	Experimento2
21 → 34	10	56	320	1,00	2,00
33 → 31	10	72	320	1,30	2,50
27 → 29	10	88	320	0,50	1,00
32 → 30	10	117	320	0,40	0,80

A topologia e as taxas de transmissão foram escolhidas de forma a dar ênfase ao parâmetro da taxa de uso dos nós. Portanto, na condução dos testes foi usado 0,80 e 0,40 respectivamente para a P_{ur} e P_m .

Os resultados para o Experimento1 e Experimento2 são apresentados respectivamente nas Tabelas 7 e 8.

Tabela 7: Resultados para o Experimento1 do Cenário 3

CA-AODV				
modo de uso	PDR	razão de perdas	atraso médio	n_{min}
mecanismo de análise desabilitado	0,63092 (0,59938 - 0,66247)	0,22626 (0,21495 - 0,23758)	0,03932 (0,03735 - 0,04128)	36,99 \approx 37
mecanismo de análise habilitado + mecanismo de correção de <i>loops</i>	0,63017 (0,59866 - 0,66168)	0,23423 (0,22252 - 0,24594)	0,03961 (0,03763 - 0,04159)	32,55 \approx 33
mecanismo de análise habilitado + mecanismo de prevenção de <i>loops</i>	0,63266 (0,60103 - 0,66429)	0,23409 (0,22239 - 0,24580)	0,03883 (0,03689 - 0,04077)	30,74 \approx 31
AODV				
	0,62187 (0,59078 - 0,65296)	0,22928 (0,21782 - 0,24075)	0,03973 (0,03774 - 0,04172)	21,14 \approx 22

Tabela 8: Resultados para o Experimento2 do Cenário 3

CA-AODV				
modo de uso	PDR	razão de perdas	atraso médio	n_{min}
mecanismo de análise desabilitado	0,39899 (0,39212 - 0,43339)	0,39890 (0,37649 - 0,41612)	0,05295 (0,05030 - 0,05560)	161,10 \approx 162
mecanismo de análise habilitado + mecanismo de correção de <i>loops</i>	0,41275 (0,39212 - 0,43339)	0,39630 (0,37649 - 0,41612)	0,05168 (0,04910 - 0,05426)	113,83 \approx 114
mecanismo de análise habilitado + mecanismo de prevenção de <i>loops</i>	0,40818 (0,38777 - 0,42859)	0,39649 (0,37667 - 0,41632)	0,05038 (0,04787 - 0,05290)	156,43 \approx 157
AODV				
	0,38994 (0,37044 - 0,40943)	0,39733 (0,37746 - 0,41719)	0,05283 (0,05019 - 0,05547)	124,16 \approx 125

Pelos resultados apresentados na Tabela 7, é visto que neste nível de tráfego não houve ganho do CA-AODV nem quanto ao próprio CA-AODV sem o mecanismo de análise nem quanto ao AODV, todos os parâmetros de medição de desempenho foram bastante próximos. Por isso, no segundo teste as taxas foram aumentadas e um ganho de desempenho é verificado.

No novo nível de tráfego usado no Experimento2, cujos resultados estão apresentados na Tabela 8, o CA-AODV apresentou ganho de desempenho tanto quando comparado ao CA-AODV sem mecanismo de análise como quando comparado ao AODV. Essa configuração não só significa mais carga como também uma maior discrepância nas taxas das fontes de dados.

O CA-AODV, usando o mecanismo corretivo contra *loops*, deu ganho de $\approx 3,3\%$ para o PDR quando comparado ao CA-AODV sem mecanismo de análise, e em relação ao AODV o ganho é de $\approx 5,8\%$.

Quando usando mecanismo preventivo contra *loops*, também é verificado ganho de desempenho para o PDR tanto quando o CA-AODV é comparado ao CA-AODV sem

mecanismo de análise como quando comparado ao AODV. No entanto, devido sua restrição na análise das rotas alternativas é visto que o emprego do mecanismo corretivo forneceu resultados melhores.

O desempenho em todos os casos foi similar quanto aos demais parâmetros de análise, razão de perdas e atraso médio.

6.1.4 Cenário 4

Dentre os cenários apresentados, este cenário é o que apresenta uma topologia mais genérica dos nós, como pode ser visto na Figura 20.

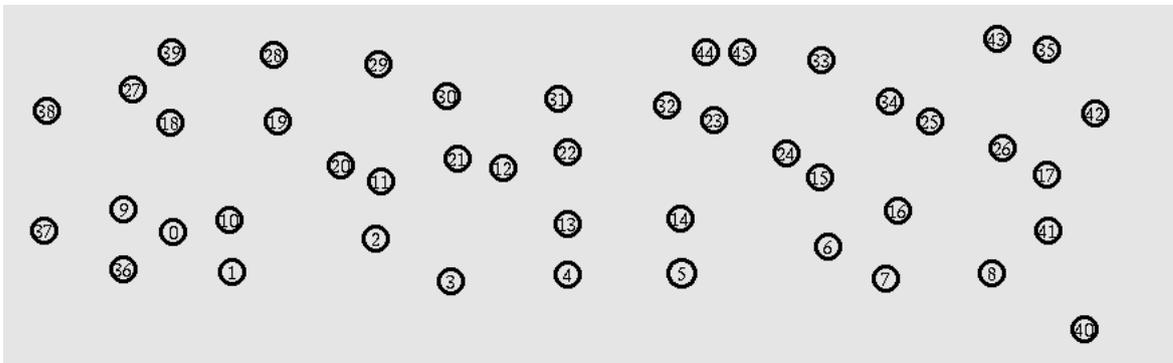


Figura 20: Quarto cenário de testes

O CA-AODV tem seu ganho desempenho mais evidenciado quando as rotas existentes na rede tendem a ficar desbalanceadas em termos de distribuição de carga. Assim, o experimento foi conduzido segundo a configuração apresentada na Tabela 9.

Tabela 9: Configuração do Cenário 4

conexão	carga útil (<i>bytes</i>)	taxa (kbps)	instante início (s)	instante fim (s)
39 → 40	10	1,00	56	320
35 → 36	10	1,30	72	320
38 → 42	10	0,50	88	320
35 → 20	10	0,50	100	320
28 → 7	10	0,40	117	320

As taxas foram propositalmente discrepantes para privilegiar a atuação do CA-AODV na melhor distribuição de carga na rede, assim foi usado 0,4 e 0,7 respectivamente para os fatores de ponderação P_m e P_{ur} . Os resultados encontrados são apresentados na Tabela 10.

Tabela 10: Resultados para o Cenário 4

CA-AODV				
modo de uso	PDR	razão de perdas	atraso médio	n_{min}
mecanismo de análise desabilitado	0,56459 (0,53636 - 0,59282)	0,23588 (0,22409 - 0,24767)	0,06004 (0,05704 - 0,06304)	11,25 \approx 12
mecanismo de análise habilitado + mecanismo de correção de <i>loops</i>	0,58375 (0,55456 - 0,61294)	0,23538 (0,22362 - 0,24715)	0,06198 (0,05888 - 0,06508)	12,22 \approx 13
mecanismo de análise habilitado + mecanismo de prevenção de <i>loops</i>	0,57145 (0,54288 - 0,60003)	0,24200 (0,22991 - 0,25411)	0,05971 (0,05673 - 0,06270)	7,71 \approx 8
AODV				
	0,56132 (0,53325 - 0,58938)	0,23591 (0,22412 - 0,24771)	0,06011 (0,05711 - 0,06312)	12,75 \approx 13

O CA-AODV, usando o mecanismo corretivo contra *loops*, deu ganho de $\approx 3,4\%$ para o PDR quando comparado ao CA-AODV sem mecanismo de análise, e em relação ao AODV o ganho é de $\approx 4,0\%$.

Quando usando mecanismo preventivo contra *loops*, também é verificado ganho de desempenho para o PDR tanto quando o CA-AODV é comparado ao CA-AODV sem mecanismo de análise como quando comparado ao AODV, em ambos os casos o ganho de desempenho é de $\approx 1,5\%$. Esse valor mostra que devido sua restrição na análise das rotas alternativas o emprego do mecanismo corretivo forneceu resultados melhores.

Quanto aos demais parâmetros de análise, razão de perdas e atraso médio, o desempenho em todos os casos foram similares.

6.2 Conclusão

Os cenários 1 e 2 foram concebidos de forma a enfatizar os pontos fortes de seleção de rotas do CA-AODV, chegando assim ao ganho de desempenho visto. Os resultados para o cenário 1, mostram claramente que em casos nos quais há interferência no meio afetando a rede de forma não homogênea a atuação do CA-AODV se dá positivamente, incrementando o desempenho da rede. No cenário 2, é verificada a eficácia do CA-AODV em distribuir melhor a carga na rede e o ganho de desempenho causado por essa distribuição.

Os cenários 3 e 4, um pouco mais complexos que os outros dois, também validam as expectativas de ganho de desempenho ao promover um equilíbrio maior na rede. Esse equilíbrio de carga na rede traz outro benefício além do desempenho, já analisado, que é a distribuição do consumo de energia dos nós na rede. Essa distribuição do consumo de energia tem impacto direto na vida útil dos nós e da rede. Para o cenário 3 o ganho

de desempenho do CA-AODV apenas foi verificado com o aumento do fluxo de dados na rede, enquanto que para o cenário 4 não foi necessário aumentar tanto a taxa de dados para alcançar benefícios ao distribuir melhor a carga na rede.

O PDR foi o parâmetro de desempenho com ganho mais acentuado, em exceção no cenário 2 onde a razão de perdas obteve melhor ganho de desempenho. O ganho de desempenho do PDR, é também o ganho que esteve presente em todos os cenários. Esse que é em geral o principal parâmetro de análise de desempenho de protocolos de roteamento [4], pois numa rede o principal componente é a informação.

Para o PDR, os resultados de ganho de desempenho ficaram entre aproximadamente 3,0% e 6,0%. Quantificando esses valores para um cenário no qual o total de envios fossem de cem mil pacotes (100.000), e caso o PDR para o AODV fosse de 50,0% teria-se a recepção de cinquenta mil pacotes. Caso fosse usado o CA-AODV, devido ao ganho de desempenho do PDR a quantidade de pacotes recebidos a mais ficaria entre 1.500 e 3.000, num universo de 50.000 pacotes originalmente recebidos no AODV.

Quanto aos mecanismos de prevenção e correção *loops*, verificou-se, como esperado, que a restrição imposta ao universo de rotas de análise pelo mecanismo de prevenção diminuiu seu ganho de desempenho em relação ao emprego do mecanismo de correção. Contudo isso não descarta o uso do mecanismo de prevenção, pois seu uso também proporcionou ganho de desempenho.

O CA-AODV sem o uso de qualquer mecanismo de análise, apresentou para todos os cenários um desempenho próximo do AODV. Como CA-AODV, com prevenção ou correção de *loops*, sempre promoveu ganho de desempenho em relação ao CA-AODV sem mecanismo de análise, pode-se concluir que o CA-AODV potencialmente tem um desempenho superior ao AODV para cenários e condições similares aos apresentados neste capítulo.

7 Implementação e Testes do CA-AODV numa Plataforma Real

Além dos testes do CA-AODV via simulador também foi avaliada uma construção desse protocolo numa plataforma real, utilizando como dispositivos sem fio os módulos XBee da *Digi*[®]. Além dos módulos XBee, foram usados microcontroladores da família ADuC8XX da *Analog Devices, Inc.* Cada nó sensor é composto de um microcontrolador, no qual o CA-AODV é implementado, mais um módulo XBee. O microcontrolador e o módulo XBee são interconectados via suas respectivas interfaces UART (*Universal Asynchronous Receiver/Transmitter*).

7.1 Módulo XBee

Esse módulo, fabricado pela *Digi International* e mostrado na Figura 21, oferece suporte para ser usado com um *firmware* padrão com o qual o módulo funciona como escravo, controlado por outro dispositivo, ou modificando uma aplicação padrão fornecida pela *Digi*[®]. Usando a aplicação padrão o módulo fica autônomo não necessitando de um outro dispositivo para gerenciar seu funcionamento.



Figura 21: Módulo XBee

O módulo possui as seguintes características:

- alcance (ambientes abertos/ambientes fechados): até 100/30 m;
- potência máxima de transmissão: 1 mW (0 dBm);
- frequência de operação: 2,4 GHz.

O núcleo do módulo é composto basicamente por um microcontrolador MC9S08GT60 e um transceptor¹ de rádio frequência MC13139, ambos da *Freescale*[®].

O microcontrolador MC9S08GT60 possui dois *timers*. Ao usar a aplicação padrão apenas um desses *timers* está disponível, pois um *timer* já é usado nas operações da aplicação padrão na UART. Como o CA-AODV tem seu funcionamento dependente da utilização de *timers*, para tomada da taxa de uso dos nós e atraso de ACK, optou-se por usar o *firmware* padrão no módulo XBee, fazendo necessário o uso de um outro microcontrolador ou um computador para gerenciar as tarefas de mais alto nível.

O *firmware* padrão possui dois modos de operação: 1) modo transparente; e 2) modo de comandos API (*Application Programming Interface*). No modo transparente o módulo atua como uma serial emulada, ou seja, todos os dados que chegam a UART são enviados em rádio frequência (RF). No modo de comandos API os dados ou comandos enviados ao módulo ou recebidos desse seguem em *frames* num formato definido.

Os *frames* enviados ao módulo incluem:

- identificador do comando;
- dados.

Os *frames* recebidos do módulo incluem:

- identificador do comando;
- evento que gerou a resposta do módulo;
- dados referentes ao evento.

O *frame* é representado da seguinte forma:

palavra de início de <i>frame</i>	comprimento do <i>frame</i>		conteúdo do <i>frame</i>		soma de verificação
0x7E	MSB	LSB	identificador do comando	argumentos	1 <i>byte</i>

¹Dispositivo que combina um transmissor e um receptor de RF utilizando componentes comuns para ambas funções

Onde, MSB e LSB são respectivamente os *bytes* mais e menos significativos do “comprimento do *frame*” e a “soma de verificação” é usada para verificar a integridade do *frame*. Para *frames* enviados ao módulo, o “conteúdo do *frame*” define o comando e se necessário os argumentos desse comando. Para *frames* recebidos do módulo, o “conteúdo do *frame*” informa o evento e o comando ao qual se refere e, caso haja, dados relacionados ao evento.

A implementação do CA-AODV foi projetada para trabalhar com os módulos no modo de comandos API, pois nesse modo a interação entre a aplicação e o módulo é maior, por meio dos comandos enviados ao módulo e suas respostas.

Para o CA-AODV, o comando de transmissão de dados ao módulo e sua resposta informando o *status* da operação são de suma importância no seu funcionamento, tanto na tomada do atraso de ACK como na manutenção da rota. O evento de recepção de pacote de RF, que gera um *frame* resposta do módulo ao microcontrolador, é usado para obter a taxa de uso dos nós.

7.2 ADuC8XX

O microcontroladores da família ADuC8XX são baseados na arquitetura do 8052. Essa arquitetura é baseada no núcleo do 8051², porém com uma memória de dados expandida para 256 *bytes* e um *timer* de 16 *bits* a mais (*timer* 2). Essa família de microcontroladores é destinada a sistemas de instrumentação.

Com microcontroladores da família ADuCXX, é possível implementar sistemas embarcados (*embedded*) com alto grau de confiabilidade. Esses microcontroladores possuem uma série de recursos que possibilitam sua aplicação em várias modalidades de instrumentação, podendo desta forma, serem utilizados para desenvolver desde sistemas de controle integrados a sensores inteligentes. Por possuir um núcleo com a arquitetura do 8052 a programação fica mais acessível, pois compiladores em linguagem C, como o KEIL [39], já são compatíveis e dispõem de recursos para a família ADuCXX.

O modelo empregado na implementação do CA-AODV foi o ADuC832, esse dispositivo tem suas características apresentadas na Tabela 11.

A estratégia de uso e reaproveitamento de recursos é um importante fator de projeto, principalmente em se tratando de uma aplicação embarcada. Por isso o gerenciamento das estruturas de armazenamento necessárias para o funcionamento do protocolo, como

²Família de microcontroladores de 8 *bits* lançada pela Intel[®]

Tabela 11: Características do ADuC8XX

Especificações	
Velocidade da CPU (<i>Mega Instruction Per Second - MIPS</i>)	1, 3
Memória <i>flash</i> de Programa (<i>kbytes</i>)	62
Memória RAM (<i>kbytes</i>)	4
Memória EEPROM de dados não volátil (<i>bytes</i>)	2304
Resolução do Conversor A/D (<i>bits</i>)	12
Velocidade do Conversor A/D (<i>Samples Per Second - SPS</i>)	247 k

as tabelas de roteamento e as tabelas que guardam as informações do pacotes RREQ recebidos, deve ser feito de maneira racional. Uma política inapropriada de utilização poderia limitar a quantidade de rotas ativas num dado nó devido a falta de memória no dispositivo.

7.3 Implementação do CA-AODV

A implementação seguiu a ideia de haver três tipos de dispositivos: nós fonte, nós destino e nós roteadores. Os nós fonte iniciam a busca por um nó destino, enviando pacote RREQ, e após a rota está ativa, receber o pacote RREP, pacotes de dados são enviados a uma taxa constante. Os nós destino ficam a espera de requisição de rota, pacote RREQ, para gerar o pacote de resposta, RREP, e ficar recebendo os pacotes de dados. Os nós roteadores têm a função de intermediar a comunicação entre os nós fonte e destino, ou seja, tem função de repassar pacotes sejam esses de dados ou de controle do protocolo.

Apenas a aplicação dos nós roteadores foi feita para o ADuC832, as demais, aplicação dos nós fonte e dos nós destino, foram feitas para serem executadas num computador. Tanto a aplicação fonte como destino foram feitas em C# (*C Sharp*) usando o *Microsoft Visual Studio* como ferramenta de desenvolvimento, enquanto que a aplicação dos nós roteadores foi escrita em linguagem C e empregou o compilador KEIL. Assim a Figura 22a representa o esquema de ligação dos nós roteadores e a Figura 22b representa o esquema de ligação dos nós fonte e destino.

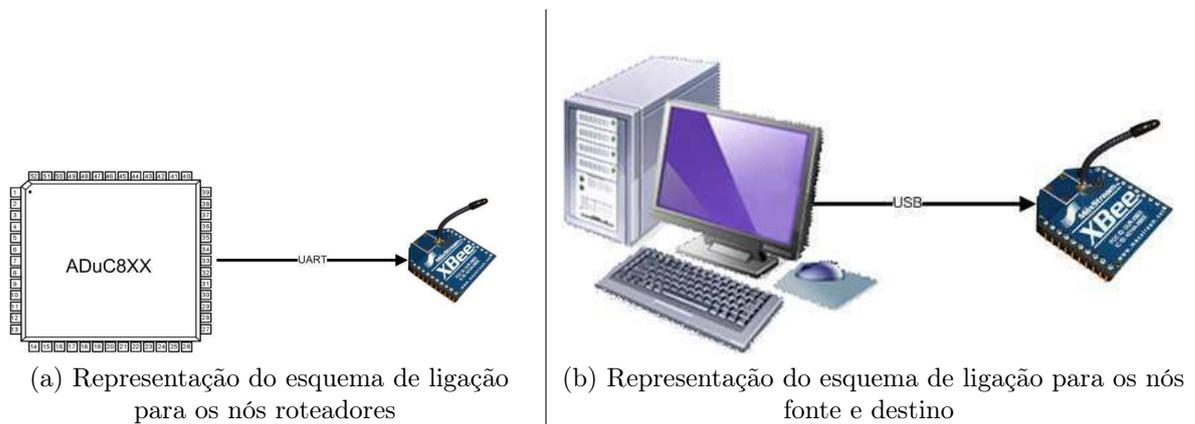


Figura 22: Esquema de ligação da implementação

7.3.1 Nós Fonte

Esses nós trabalham com todos os tipos de pacotes controle (RREQ, RREP, ERROR) e dados. Sendo que os pacotes RREQ e de dados são apenas gerados, para transmissão, e os pacotes RREP e ERROR apenas recebidos e tratados. Assim essa aplicação deve descartar pacotes recebidos do tipo RREQ e de dados.

Na Figura 23 são apresentados os fluxogramas que representam o funcionamento da aplicação para os nós fonte. Essa aplicação possui dois estados: 1) estado de requisição de rota; e 2) estado de transmissão de dados. No estado de requisição, mostrado na Figura 23b, a aplicação fonte está no ciclo de construção de nova rota. Ao terminar o estado de requisição, receber pacote RREP para o destino desejado, a aplicação entra no estado de transmissão de dados, mostrado na Figura 23d. Como é observado na Figura 23c apenas pacotes RREP e ERROR são tratados pela nó fonte, os demais são descartados.

No estado de requisição de rota cada pacote RREQ enviado tem um tempo expiração, no qual a tentativa de encontrar a rota desejada é considerada falha e um novo RREQ é enviado. Esse tempo de expiração é representado pelo bloco de decisão de RREQ pendente, na Figura 23b.

O envio de dados é feito como apresenta o esquema da Figura 23d, onde uma nova transmissão só é agendada quando não há envio de pacote pendente (agendado).

O estado inicial do programa é de requisição e uma vez construída rota até seu respectivo destino o estado de transmissão é assumido.

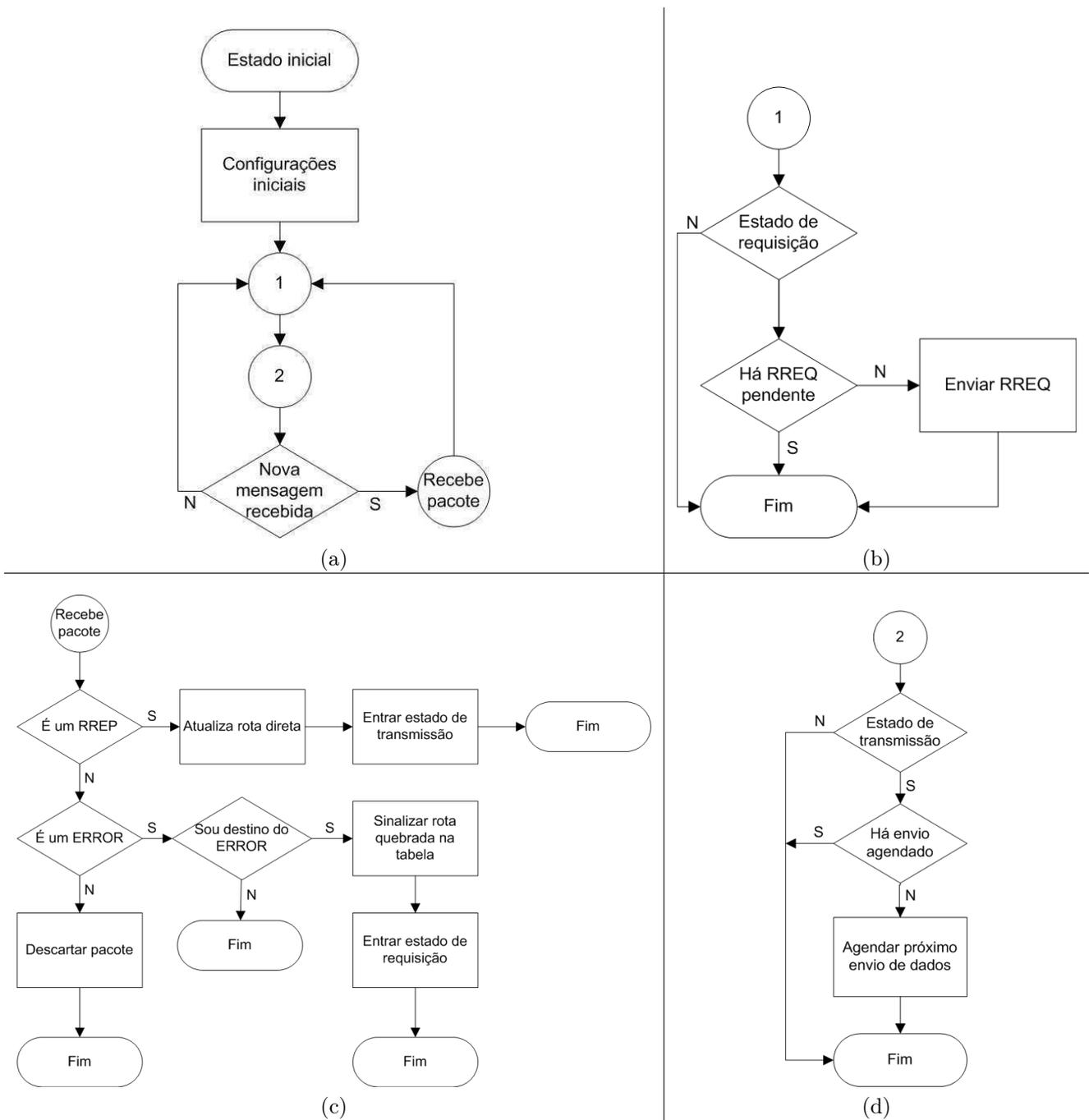


Figura 23: Fluxo de código para o nó fonte

7.3.2 Nós Destino

Os nós destino são inicializados e ficam a espera de pacotes RREQ ou de dados que lhes sejam endereçados. Ao receber um pacote RREQ o nó verifica qual o destino desejado, sendo seu o endereço requisitado um RREP é gerado. Esta aplicação trata com três tipos de pacotes: RREQ, RREP e de dados. Os pacotes de RREQ e dados são para recepção e RREP para transmissão. Qualquer pacote recebido que não seja RREQ ou de dados deve

ser descartado.

Na figura 24 é apresentado o fluxo de código da aplicação destino.

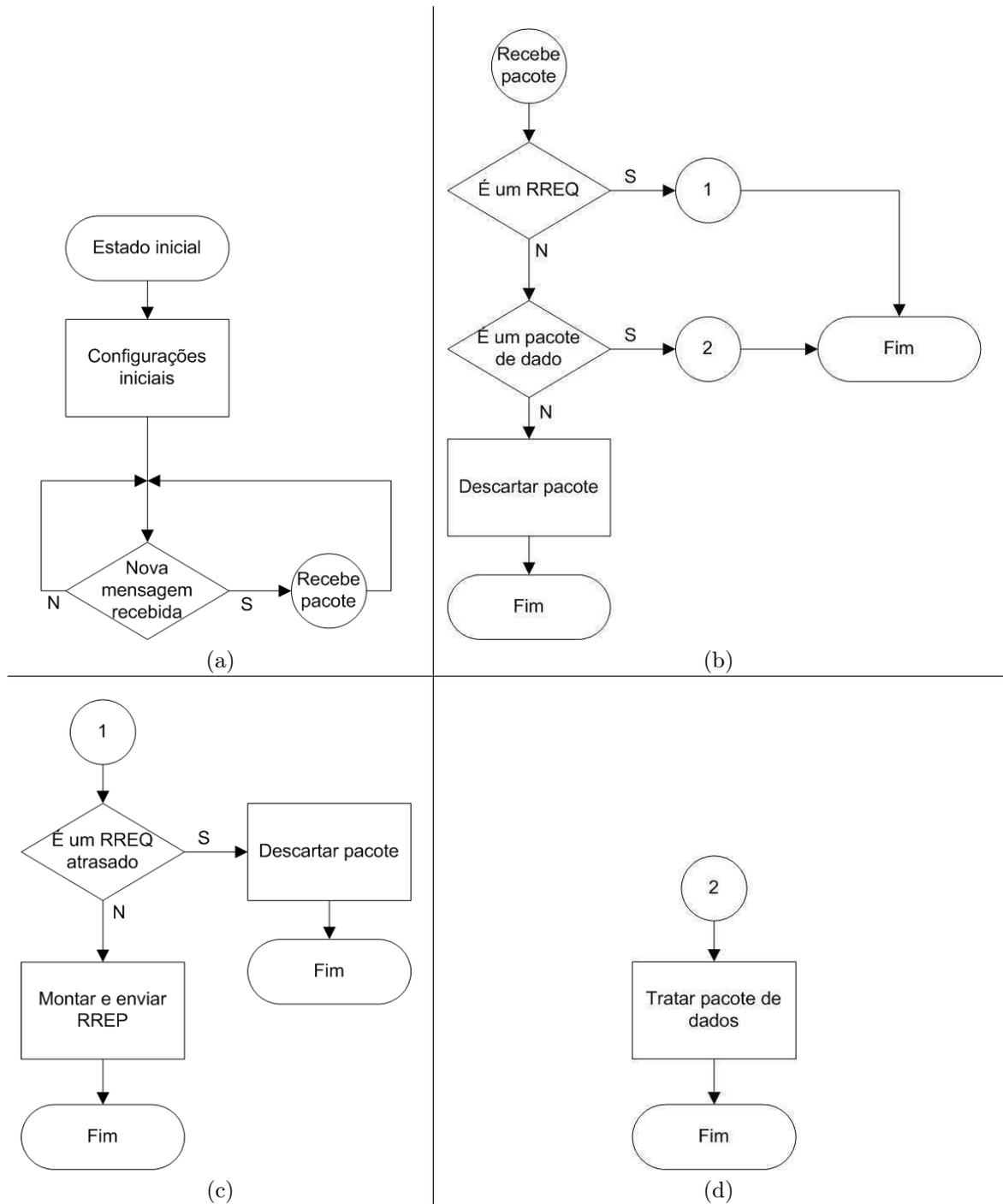


Figura 24: Fluxo de código para o nó destino

7.3.3 Nós Roteadores

Este programa deve tratar todos os tipos de pacotes de controle (RREQ, RREP e ERROR) e dados. Os pacotes de RREQ são passíveis apenas para repasse, ao receber um RREQ esse é processado para decidir se o repassa ou descarta-o. Os pacotes de dados também são para repasse apenas, pacotes de dados recebidos para rotas que ainda não existem são descartados. Os pacotes RREP são para repasse, recebe e processa e então o repassa ou descarta-o, contudo esses pacotes também podem ser gerados no nó roteador caso esse já possua informação de rota direta recente até o destino. Os pacotes ERROR são para repasse até o nó fonte, acompanhado de processamento atualização da tabela de roteamento. Assim como os pacotes RREP, os pacotes ERROR podem ser gerados no nó roteador.

Muitas tarefa da aplicação fazem uso dos *timers* do ADuC. Usa-se *timer* para tomar as métricas de disponibilidade (atraso de ACK e taxa de uso) e para fazer a limpeza da tabela de roteamento das rotas inativas.

A aplicação do nó roteador, no ADuC, fica sempre a espera de uma nova mensagem enviada pelo módulo XBee via UART, seja essa mensagem um pacote de RF recebido ou alguma resposta do módulo ao envio de algum comando por parte da aplicação. Caso seja um pacote de RF, esse pode ser algum pacote de controle do protocolo ou um pacote de dados para repasse até seu destino. Dependendo do tipo de pacote de controle diferentes procedimentos são tomados para sua recepção. Essa discussão é representada pelo fluxo de código mostrado na Figura 25.

Ao receber um pacote um novo RREQ o nó roteador:

- guarda informações para identificação do RREQ (BID, endereço do nó fonte, instante de recebimento);
- configura a rota reversa; e
- verifica se possui a rota direta até o destino:
 - caso possua um RREP é gerado;
 - caso contrário o RREQ é repassado.

Antes do repasse do RREQ, esse tem seus campos atualizados com as informações de métrica do roteador em questão.

Ao receber um pacote RREQ “atrasado”:

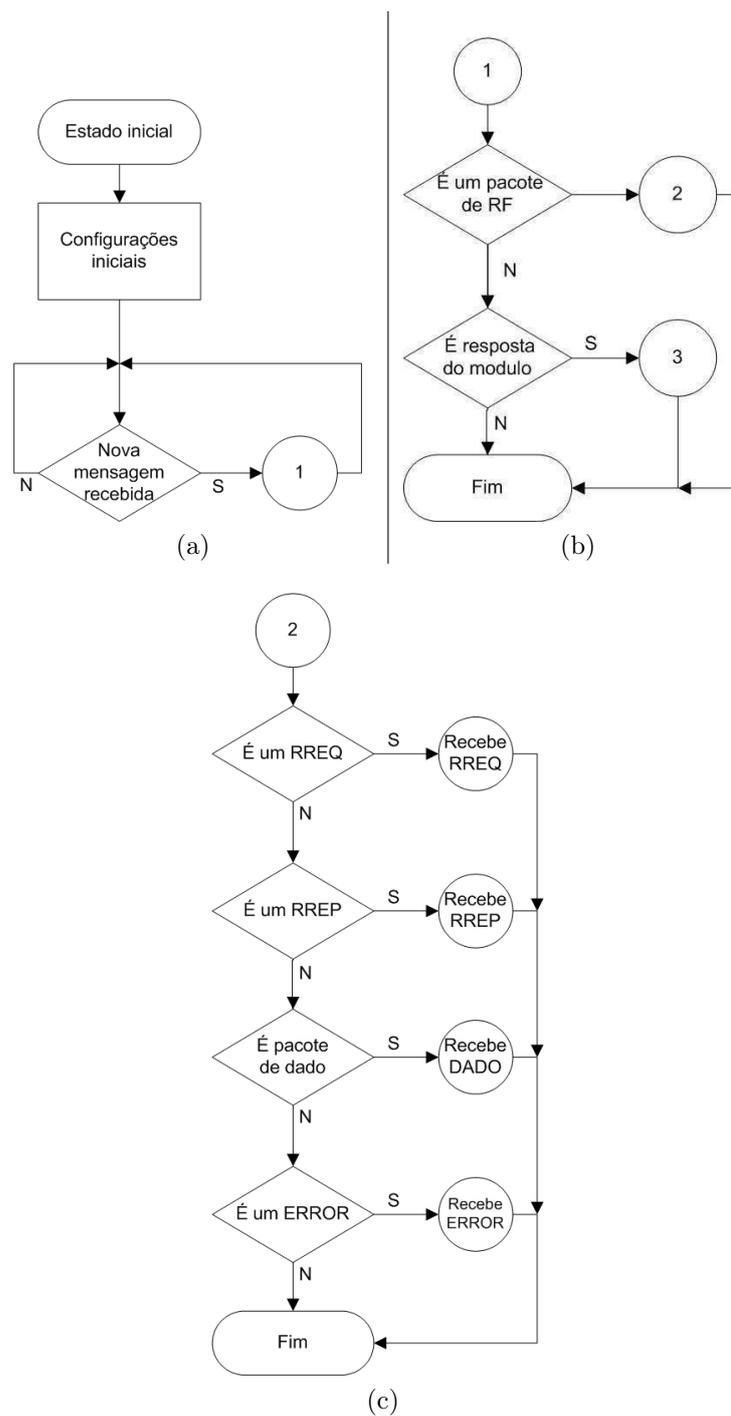


Figura 25: Fluxo de código para o nó roteador na recepção de pacotes via UART

- o nó roteador verifica se ainda trata do RREQ “atrasado”
- caso trate:
 - as rotas reversas, antiga na tabela de roteamento e nova vinda no RREQ, são comparadas.

- há repasse dos RREQ “atrasados” que foram aceitos.
- caso contrário o RREQ é descartado.

Ao receber um pacote RREP:

- caso seja um RREP mais recente:
 - a rota direta é atualizada;
 - o pacote é retransmitido
- caso não seja novo:
 - é verificada a formação de *loop*.
 - caso o loop seja detectado:
 - * as medidas de correção de *loop* são tomadas
 - caso não seja detectado loop:
 - * o RREP é descartado.

Na Figura 26 é apresentado o diagrama de fluxo da recepção dos pacotes descritos acima, RREQ e RREP.

A medida de disponibilidade é tomada sobre os nós roteadores. Ao receber um pacote de dados, o nó roteador o repassa caso exista rota direta e caso contrário o descarta. Ao repassar um pacote de dados o nó guarda as informações de identificação do pacote (ID) e o do instante de envio. O atraso de ACK é obtido a partir da resposta do módulo XBee. O XBee na transmissão *unicast* gera um pacote de resposta ao envio informando o *status* da transmissão, sucesso ou falha. Essa resposta é gerada na recepção do pacote de ACK (sucesso na transmissão) ou ao fim do tempo de espera de resposta por parte do módulo de destino (falha na transmissão).

Quanto à taxa de uso, sua tomada é feita na recepção dos pacotes de dados. O instante de recebimento de cada pacote de dado é guardado numa pilha, e quando necessária a taxa de uso toma-se o intervalo de tempo entre as últimas dez recepções de pacotes de dados e obtém-se sua taxa de uso. Essa quantidade de dez pacotes para tomada da taxa foi escolhida por simples decisão de projeto.

Esses dois procedimentos, descritos acima, para tomada do atraso de ACK e da taxa de uso são executados a partir do recebimento de pacotes vindos do módulo XBee. Ao

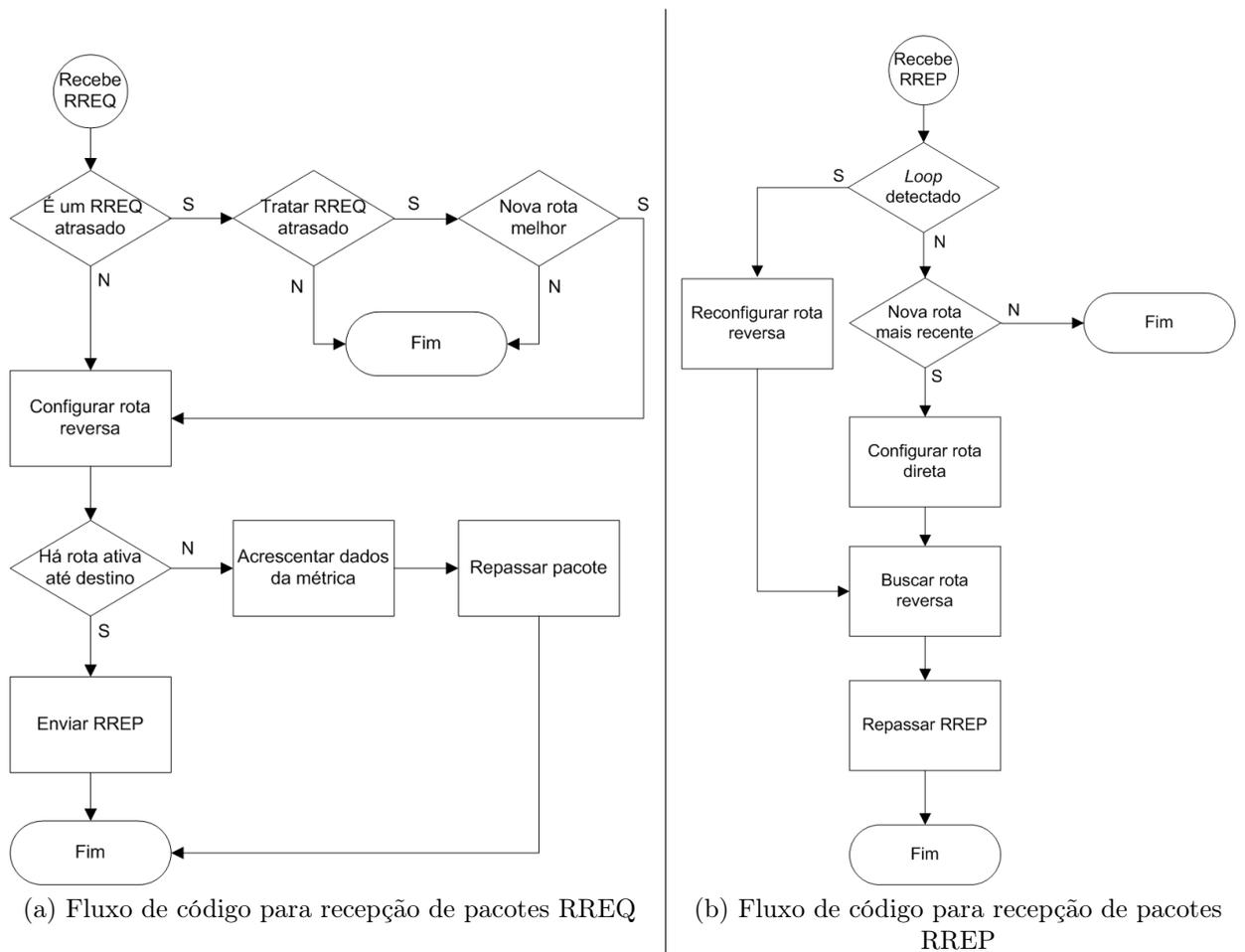


Figura 26: Fluxo de código para o nó roteador na recepção de pacotes RREQ e RREP

receber algum pacote vindo do módulo a aplicação verifica o tipo do pacote. Caso seja uma pacote de RF e seja um pacote de dados, são atualizadas os dados para cálculo da taxa de uso. Caso seja uma resposta do módulo ao envio de um pacote dados, as informações para a tomada do atraso de ACK são atualizadas. Essa verificação é mostrada no fluxograma apresentado na Figura 27.

A verificação de falha na transmissão faz parte da monitoração da rota. Na ocorrência de consecutivas falhas na transmissão haverá sinalização de rota “quebrada”, como é mostrado na Figura 27a. Uma rota sinalizada como “quebrada” ao receber um pacote de dados gera um pacote ERROR e o encaminha aos nós que usam tal rota inativa, conforme é visto no fluxo de código apresentado na Figura 27b. Ao longo do percurso do pacote ERROR até o nó fonte, os nós roteadores devem atualizar suas informações quanto a rota inativa e repassar o pacote ERROR para que esse chegue até o respectivo nó fonte de destino, com é visto na representação do código apresentado na Figura 28.

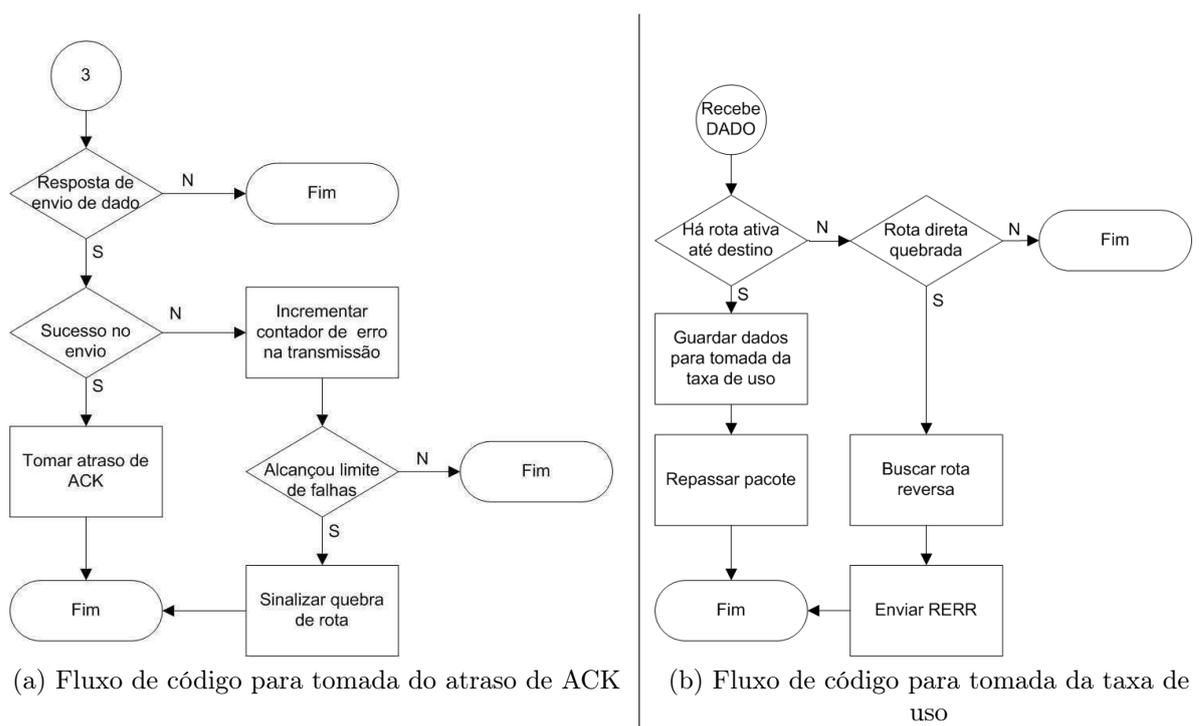


Figura 27: Fluxo de código para o nó roteador para tomada das métricas de disponibilidade

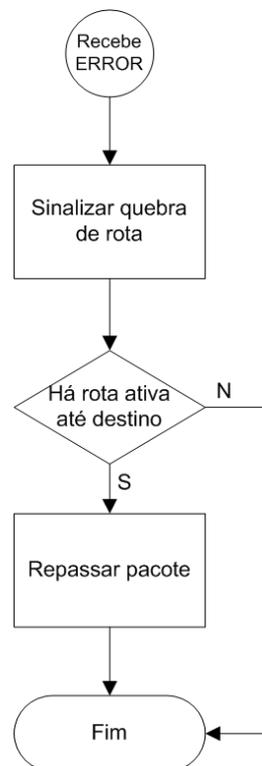


Figura 28: Fluxo de código para o nó roteador na recepção de pacote ERROR

7.4 Testes da Implementação

Os testes da implementação foram feitos no edifício Gurdip Singh Deep do Centro de Engenharia Elétrica e Informática da UFCG. Na Figura 29 é apresentada uma vista superior de como foram alocados os nós (fonte, destino e roteadores) em suas respectivas salas.

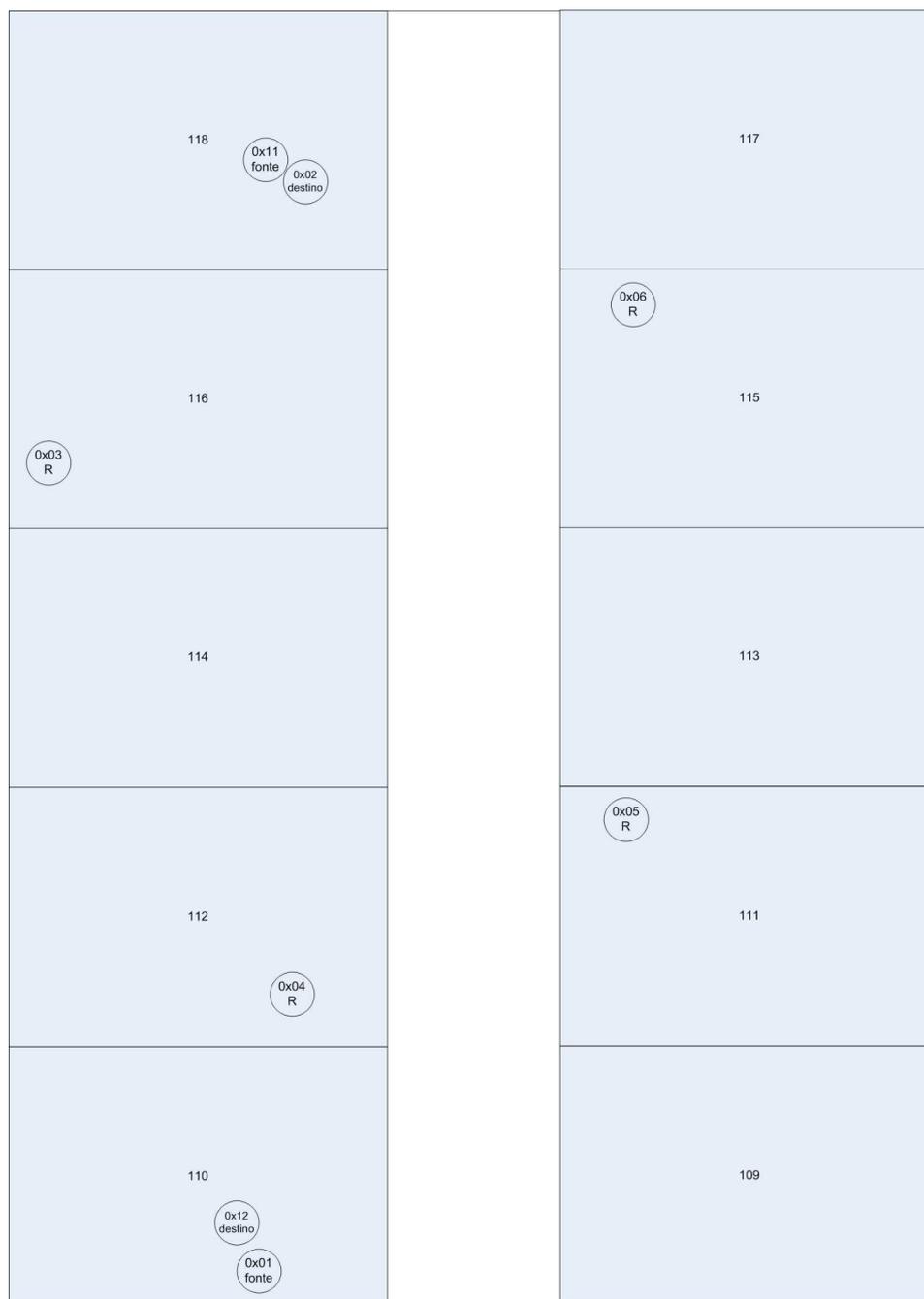


Figura 29: Vista superior do cenário de testes

O cenário foi montado com dois nós fonte, endereçados como “0x01” e “0x11”, e dois

nós destino, endereçados como “0x02” e “0x12”. Os demais nós (0x03, 0x04, 0x05 e 0x06) são roteadores. Os nós foram alocados e suas potências configuradas de forma a haver basicamente duas rotas ligando os nós fonte e destino (“fonte/destino \leftrightarrow 0x04 \leftrightarrow 0x03 \leftrightarrow destino/fonte” e “fonte/destino \leftrightarrow 0x04 \leftrightarrow 0x05 \leftrightarrow 0x06 \leftrightarrow destino/fonte”).

Os testes foram conduzidos com primeiro o nó “0x01” começando a requisitar rota até o nó “0x02”, e após a rota ser formada o nó “0x11” faz requisição de rota até o nó “0x12”. Após a formação das rotas os nós fonte começavam a transmitir a uma taxa de 2 pacotes por segundo.

Na Tabela 12 são apresentados os resultados de dois experimentos feitos para a implementação, um no qual ocorreu a concentração de fluxo numa única rota (Experimento 1) e outro no qual as rotas ficaram distribuídas nos nós (Experimento 2).

Tabela 12: Resultados no testes da implementação CA-AODV

Experimento 1					
conexão	atraso para achar rota (s)	rota	saltos	PDR	PDR total
0x01 \rightarrow 0x02	5	0x01 \rightarrow 0x04 \rightarrow 0x03 \rightarrow 0x02	3	0,86	0,87
0x11 \rightarrow 0x12	0,5	0x11 \rightarrow 0x03 \rightarrow 0x04 \rightarrow 0x12	3	0,87	
Experimento 2					
0x01 \rightarrow 0x02	5	0x01 \rightarrow 0x04 \rightarrow 0x03 \rightarrow 0x02	3	0,89	0,91
0x11 \rightarrow 0x12	0,4	0x11 \rightarrow 0x06 \rightarrow 0x05 \rightarrow 0x04 \rightarrow 0x12	4	0,94	

Esses resultados mostram o ganho de desempenho ao se escolher uma rota menos usada mesmo quando a quantidade de saltos é maior, pois ambas conexões apresentaram um melhor desempenho no Experimento 2. O Experimento 2 apresentou um “PDR total” aproximadamente 5% superior ao do Experimento 1.

7.5 Conclusão

Os resultados encontrados para a implementação do CA-AODV mostram na prática o ganho vindo da busca por rotas menos utilizadas. Devido a tal característica também é conseguida uma maior vida útil da rede.

A implementação do CA-AODV não traz grande complexidade nem necessita de elevados requisitos de *hardware*, pois o mesmo foi implementado num microcontrolador (ADuC832) de arquitetura básica e sem necessidade de elevada capacidade de processamento.

8 Conclusões

Neste trabalho desenvolveu-se um protocolo de roteamento, voltado para redes de sensores sem fio, que faz uso de métricas de qualidade baseadas em informações extraídas das próprias conexões existentes na rede. Propôs-se um modelo diferente para as métricas de qualidade utilizadas na seleção das rotas. Esse modelo, faz uma classificação das rotas baseado em informações tomadas sob o ponto de vista dos nós e não dos enlaces. Tomar as métricas sob o ponto de vista dos nós, fez com que a métrica fosse baseada na disponibilidade de recursos dos nós pertencentes as rotas, essa disponibilidade diz respeito a recursos de *hardware*, do nó em questão, e do nível de congestionamento do meio no qual esse nó se encontra. Assim as métricas escolhidas para análise das rotas foram: 1) atraso de ACK; e 2) taxa de uso dos nós pertencentes à rota. Esse modelo de análise das rotas tem como vantagem o emprego da informação de disponibilidade dos nós para diferentes rotas, ou seja, nesse modelo a métrica de análise das rotas possui uma maior aplicabilidade do que usar informações sobre os enlaces de dados existentes.

O ambiente para desenvolvimento e testes do protocolo usado foi o NS-2, um simulador de redes bastante difundido no meio acadêmico e com suporte para vários tipos de redes e de tráfego. Uma das vantagens do NS-2 é sua flexibilidade quanto à modificação de seus blocos para as necessidades do usuário, pois esse é um simulador de código aberto com muitas fontes de informação e trabalhos desenvolvidos.

O modelo de protocolo usado como base foi o AODV, um protocolo bastante estudado e usado em redes de sensores sem fio. Esse protocolo reativo apresenta características importantes como baixa carga de pacotes de controle e ausência de *loops*.

O CA-AODV usa informações das rotas existentes para construir novas rotas, portanto, o mecanismo tem efeito no desempenho da rede ao passo que novas rotas comecem a ser requisitadas e entrem em operação.

Os resultados, via simulação apresentados no Capítulo 6, que comparam o desempenho do CA-AODV com o CA-AODV sem uso do mecanismo de análise de rotas e com o AODV,

comprovam o ganho de desempenho na escolha por rotas menos congestionadas. Esses resultados corroboram o uso das métricas propostas para análise de disponibilidade das rotas, assim como o modelo empregado para composição do atraso de ACK e da taxa de uso dos nós numa única métrica de disponibilidade, com a qual toma-se a decisão de qual rota tomar.

O que o mecanismo de análise de rotas do CA-AODV faz é buscar rotas menos congestionadas, seja esse congestionamento pertinente ao grau de uso dos nós na rota ou ao uso do meio. Fazer opção por uma rota menos congestionada realmente proporciona ganho de desempenho, como visto pelos resultados. No entanto, o fato do CA-AODV buscar rotas com tais características de congestionamento mínimo não significa que sempre será encontrada a rota com o mínimo congestionamento nem que essa rota proporcionará acréscimo de ganho desejado. Assim, também é necessário que esse ganho proporcionado pelo modelo de análise tenha repetibilidade para um mesmo cenário, assim garante-se que esse ganho de desempenho não é um caso isolado num universo grande de experimentos. A análise estatística dos resultados para avaliação dos protocolos, apresentada na Seção 3.3, tem o intuito de verificar exatamente o quão recorrente é o ganho de desempenho proporcionado pela modificação. Baseado nos resultados apresentados no Capítulo 6, comprova-se que não só o CA-AODV proporciona ganho de desempenho, ao escolher rotas menos congestionadas, como também esse ganho apresenta-se de maneira recorrente ao repetir-se o experimento.

Dos parâmetros de análise dos protocolos (razão de entrega de pacotes – PDR, atraso médio dos pacotes e razão de perda de pacotes), verificou-se que o PDR é o parâmetro com maior sensibilidade à escolha de rotas menos congestionadas, estando o ganho de desempenho desse parâmetro presente em todos os cenários analisados. O atraso médio de chegada dos pacotes de dados apresentou resultados quase que indiferentes a escolha por rotas menos congestionadas e a razão de perdas apresentou ganho de desempenho em alguns cenários e em outros não. Sendo a informação o foco principal de qualquer rede, configura-se o PDR como principal fator de análise de desempenho.

Como visto pelos resultados do Capítulo 6, o CA-AODV tem seu ganho de desempenho evidenciado em condições nas quais haja possibilidade de desequilíbrio no congestionamento do meio e na distribuição de carga nos nós da rede, conexões com diferentes taxas de transmissão, ou na ação de uma interferência não homogênea sobre a rede. Uma consequência interessante da distribuição da carga nos nós da rede é a uniformidade no consumo de energia desses nós.

No Capítulo 7, foram apresentados os detalhes da implementação do CA-AODV numa plataforma real. Foi verificado, via os testes feitos para a plataforma, o ganho de desempenho que a escolha por rotas menos congestionadas proporciona ao sistema, mesmo quando escolhe-se uma rota com maior número de saltos.

8.1 Sugestões para Trabalhos Futuros

Como estudos complementares que podem dar sequência a este trabalho, destacam-se:

- Em relação a complementações e modificação no modelo do CA-AODV, pode-se:
 - Investigar diferentes métodos para composição da métrica de disponibilidade além dos apresentados neste trabalho, as Equações 5.5-5.6 e o modelo *fuzzy*;
 - Modificar o CA-AODV de forma que o mecanismo de análise por meio das métricas de congestionamento dos nós não atue somente na construção das rotas, mas também na sua manutenção podendo assim atuar na degradação da rota e não somente na sua perda;
 - Usar, além das métricas de disponibilidade propostas, métricas relacionadas a qualidade do enlace, como nível do sinal recebido, relação sinal ruído, etc. Essa métrica relacionada ao enlace poderia ser obtida a partir da recepção dos pacotes RREQ, tratando assim de forma mais seletiva as rotas passíveis de análise.
- Investigar o uso das métricas de disponibilidade apresentadas neste trabalho em outros modelos de protocolo, além do AODV estudado neste trabalho, como o DSDV [40], etc.

Referências

- [1] BARBEAU, M.; KRANAKIS, E. *Principles of Ad Hoc Networking*. West Sussex PO19 8SQ, England: John Wiley & Sons Ltd, 2007.
- [2] MAHALIK, N. P. *Sensor Networks and Configuration*. [S.l.]: Springer, 2007.
- [3] GUNASEKARAN, R.; QI, H. Xlrp: Cross layer routing protocol for wireless sensor networks. *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, p. 2135–2140, 31 2008-April 3 2008. ISSN 1525-3511.
- [4] VEERAYYA, M. *An Energy-Aware On-Demand Routing Protocol for Ad-Hoc Wireless Networks*. Dissertação (Mestrado) — Indian Institute of Technology Bombay, July 2008.
- [5] AHMED, A. A.; FISAL, N. A real-time routing protocol with load distribution in wireless sensor networks. *Comput. Commun.*, Butterworth-Heinemann, Newton, MA, USA, v. 31, n. 14, p. 3190–3203, 2008. ISSN 0140-3664.
- [6] SENEL, M. et al. A kalman filter based link quality estimation scheme for wireless sensor networks. In: *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*. [S.l.: s.n.], 2007. p. 875–880.
- [7] KARBASCHI, G.; FLADENMULLER, A. A link-quality and congestion-aware cross layer metric for multi-hop wireless routing. *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, p. 7 pp.–, 2005.
- [8] CHEN, J. et al. Lqer: A link quality estimation based routing for wireless sensor networks. *Sensors*, v. 8, n. 2, p. 1025–1038, 2008. ISSN 1424-8220. Disponível em: <<http://www.mdpi.com/1424-8220/8/2/1025>>.
- [9] YARVIS, M. et al. *Real-world experiences with an interactive ad hoc sensor network*. 2002. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.6069>>.
- [10] PERKINS, C.; ROYER, E. Ad-hoc on-demand distance vector routing. *Mobile Computing Systems and Applications, IEEE Workshop on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 90, 1999.
- [11] SAKURAI, Y.; KATTO, J. Aodv multipath extension using source route lists with optimized route establishment. *Wireless Ad-Hoc Networks, 2004 International Workshop on*, p. 63–67, June 2004.
- [12] ISSARIYAKUL, T.; HOSSAIN, E. *Introduction to Network Simulator NS2*. 1. ed. [S.l.]: Springer, 2008.

- [13] BERKELEY LBL, U. U.; PARC, X. *The ns Manual (formerly ns Notes and Documentation)*. California, March 2008.
- [14] WITTMANN, E. *Routing in Wireless Sensor Networks*. [S.l.]: VDM Verlag Dr. Müller Aktiengesellschaft & Co KG, 2007.
- [15] HEINZELMAN, W. et al. Energy-scalable algorithms and protocols for wireless microsensor networks. *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, v. 6, p. 3722–3725 vol.6, 2000.
- [16] AKKAYA, K.; YOUNIS, M. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, v. 3, n. 3, p. 325–349, May 2005. Disponível em: <<http://dx.doi.org/10.1016/j.adhoc.2003.09.010>>.
- [17] SOCIETY, I. C. *IEEE Standard for Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks- Specific requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*. New York, Oct 2003.
- [18] JAYANT, N. *Broadband Last Mile: Access Technologies for Multimedia Communications (Signal Processing and Communications)*. [S.l.]: CRC Press, 2005.
- [19] GUTIERREZ, J. A. On the use of iee 802.15.4 to enable wireless sensor networks in building automation. In: . [s.n.], 2004. v. 3, p. 1865–1869 Vol.3. Disponível em: <<http://dx.doi.org/10.1007/s10776-007-0063-4>>.
- [20] AL-KARAKI, J. N.; KAMAL, A. E. Routing techniques in wireless sensor networks: a survey. *Wireless Communications, IEEE [see also IEEE Personal Communications]*, v. 11, n. 6, p. 6–28, 2004. Disponível em: <<http://dx.doi.org/10.1109/MWC.2004.1368893>>.
- [21] JOHNSON, D. B.; MALTZ, D. A. Dynamic source routing in ad hoc wireless networks. In: IMIELINSKI; KORTH (Ed.). *Mobile Computing*. Kluwer Academic Publishers, 1996. v. 353. Disponível em: <<http://citeseer.ist.psu.edu/johnson96dynamic.html>>.
- [22] COUTO, D. S. J. D. et al. Performance of multihop wireless networks: Shortest path is not enough. In: ACM SIGCOMM. *Proceedings of the First Workshop on Hot Topics in Networks (HotNets-I)*. Princeton, New Jersey, 2002. Disponível em: <<http://citeseer.ist.psu.edu/decouto02performance.html>>.
- [23] INTANAGONWIWAT, C. et al. Directed diffusion for wireless sensor networking. *IEEE/ACM Trans. Netw.*, IEEE Press, Piscataway, NJ, USA, v. 11, n. 1, p. 2–16, February 2003. ISSN 1063-6692. Disponível em: <<http://dx.doi.org/10.1109/TNET.2002.808417>>.
- [24] KARP, B.; KUNG, H. T. Gpsr: greedy perimeter stateless routing for wireless networks. In: *Mobile Computing and Networking*. [s.n.], 2000. p. 243–254. Disponível em: <<http://citeseer.ist.psu.edu/karp00gpsr.html>>.

- [25] MELODIA, T.; VURAN, M. C.; POMPILI, D. The State of the Art in Cross-layer Design for Wireless Sensor Networks. In: *Proceedings of EuroNGI Workshops on Wireless and Mobility. Springer Lecture Notes in Computer Science 3883*. Como, Italy: [s.n.], 2005.
- [26] KAWADIA, V.; KUMAR, P. A cautionary perspective on cross-layer design. *Wireless Communications, IEEE*, v. 12, n. 1, p. 3–11, Feb. 2005. ISSN 1536-1284.
- [27] PUNDE, J.; PISSINOU, N.; MAKKI, K. On quality of service routing in ad-hoc networks. *Local Computer Networks, 2003. LCN '03. Proceedings. 28th Annual IEEE International Conference on*, p. 276–278, Oct. 2003. ISSN 0742-1303.
- [28] HE, T. et al. Speed: A stateless protocol for real-time communication in sensor networks. In: *ICDCS '03: Proceedings of the 23rd International Conference on Distributed Computing Systems*. Washington, DC, USA: IEEE Computer Society, 2003. p. 46. ISBN 0-7695-1920-2.
- [29] AHMED, A. A. et al. Real-time routing in wireless sensor networks. *Distributed Computing Systems Workshops, International Conference on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 114–119, 2008. ISSN 1545-0678.
- [30] BROCH, J. et al. A performance comparison of multi-hop wireless ad hoc network routing protocols. In: *MobiCom '98: Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*. New York, NY, USA: ACM, 1998. p. 85–97. ISBN 1-58113-035-X. Disponível em: <<http://portal.acm.org/citation.cfm?id=288256>>.
- [31] JOHANSSON, P. et al. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. In: *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. New York, NY, USA: ACM, 1999. p. 195–206. ISBN 1-58113-142-9.
- [32] CHEN, C.; MA, J. Simulation study of aodv performance over ieee 802.15.4 mac in wsn with mobile sinks. In: *AINAW '07: Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops*. Washington, DC, USA: IEEE Computer Society, 2007. p. 159–164. ISBN 0-7695-2847-3.
- [33] JAIN, R. K. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. [S.l.]: Wiley, 1991.
- [34] L'ECUYER, P. *Good Parameters And Implementations For Combined Multiple Recursive Random Number Generators*. 1998.
- [35] ADYA, A. et al. A multi-radio unification protocol for ieee 802.11 wireless networks. In: . [s.n.], 2004. p. 344–354. Disponível em: <<http://dx.doi.org/10.1109/BROADNETS.2004.8>>.
- [36] ZADEH, L. A. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, June 1965. Disponível em: <[http://dx.doi.org/10.1016/S0019-9958\(65\)90241-X](http://dx.doi.org/10.1016/S0019-9958(65)90241-X)>.
- [37] ROSS, T. J. *FUZZY LOGIC WITH ENGINEERING APPLICATIONS*. Second. West Sussex PO19 8SQ, England: John Wiley & Sons Ltd, 2004.

-
- [38] YAGER, R. *A new methodology for ordinal multiobjective decisions based on fuzzy sets*. [S.l.]: Decis. Sci., 1981. 589–600 p.
- [39] COMPANY, A. *Embedded Development Tools*. maio 2009. Disponível em: <<http://www.keil.com/>>.
- [40] PERKINS, C.; BHAGWAT, P. Highly dynamic destination-sequenced distance-vector routing (dsv) for mobile computers. In: . [S.l.: s.n.], 1994. p. 234–244.